Allen-Bradley

PanelBuilder™ 1400e Configuration Software for Windows®

(Cat. No. 2711E–ND1)

User Manual

Important User Information

Because of the variety of uses for the products described in this publication, those responsible for the application and use of this control equipment must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes and standards.

The illustrations, charts, sample programs and layout examples shown in this guide are intended solely for purposes of example. Since there are many variables and requirements associated with any particular installation, Allen-Bradley does not assume responsibility or liability (to include intellectual property liability) for actual use based upon the examples shown in this publication.

Allen-Bradley publication SGI-1.1, *Safety Guidelines for the Application, Installation, and Maintenance of Solid-State Control* (available from your local Allen-Bradley office), describes some important differences between solid-state equipment and electromechanical devices that should be taken into consideration when applying products such as those described in this publication.

Reproduction of the contents of this copyrighted publication, in whole or in part, without written permission of Allen-Bradley Company, Inc., is prohibited.

Throughout this manual we use notes to make you aware of safety considerations:

ATTENTION: Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss.

Attention statements help you to:

- identify a hazard
- avoid the hazard
- recognize the consequences

Important: Identifies information that is critical for successful application and understanding of the product.

PanelBuilder, PanelView, Data Highway Plus, DH+, ControlNet, and SLC are trademarks, and PLC, PLC-2, and PLC-3 are registered trademarks of Allen-Bradley Company, Inc.
RSView and RSLinx are trademarks of Rockwell Software, Inc.
IBM, PC, PS/2, VGA, and PC-DOS are registered trademarks of International Business Machines Corporation.
Epson is a registered trademark of Seiko Epson Corporation.
Ethernet is a registered trademark of Digital Equipment Corporation, Intel, and Xerox Corporation.
Microsoft, Windows, MS, and MS–DOS are registered trademarks of Microsoft Corporation.
Mouse Systems is a trademark of MSC Technologies, Inc.
AutoCAD is a registered trademark of Autodesk Inc.
Taylor, and ProWORXPLUS are registered trademarks of Taylor Industrial Software, Inc.
MODICON, MODSOFT, Quantum, and Modbus are registered trademarks of AEG Schneider Automation, Inc.
All other brand and product names are trademarks of their respective companies and are hereby acknowledged.

Table of Contents

Preface

Welcome to PanelBuilder 1400e Configuration Software for Windows P–1
- Windows NT Compatibility . . . P–1
- ControlNet Station Addressing . . . P–1
- Ethernet Pass-Through . . . P–1
- CD-ROM Installation . . . P–2
- RSLinx Communication Drivers . . . P–2

Terminology . . . P–2
Registering Your Copy of PanelBuilder 1400e . . . P–3
Available Documentation . . . P–3
What's in the User Manual? . . . P–4
- Preface . . . P–4
- Chapter 1, Introduction to PanelBuilder . . . P–4
- Chapter 2, Planning Applications . . . P–4
- Chapter 3, Working with Applications . . . P–4
- Chapter 4, Working with Editors . . . P–4
- Chapter 5, Defining Communications . . . P–4
- Chapter 6, Defining Tags . . . P–4
- Chapter 7, Creating Expressions . . . P–4
- Chapter 8, Creating Screens . . . P–5
- Chapter 9, Creating Objects . . . P–5
- Chapter 10, Creating Messages . . . P–5
- Chapter 11, Configuring Alarms . . . P–5
- Chapter 12, Configuring Terminal Setup Options . . . P–5
- Chapter 15, Transferring Applications . . . P–5

Who Should Read This Manual? . . . P–6
Related Publications . . . P–6
Conventions Used . . . P–7
- Print Conventions . . . P–7
- Mouse Conventions . . . P–7
- Selection Conventions . . . P–7
  - Selection Conventions in Dialog Boxes . . . P–8
- Key Conventions . . . P–9
- Command Conventions . . . P–9

Before You Begin . . . P–10
Technical Support Services . . . P–10

Introduction to PanelBuilder — Chapter 1

PanelBuilder Components 1–1
What Are You Building? 1–2
Version Compatibility 1–2
Required Equipment 1–3
- PanelBuilder Software 1–3
- Panelbuilder Transfer Utility 1–3
- PanelView Terminal 1–3
- Transferring or Installing the Application 1–4
- Programmable Controllers 1–5
- Programmable Controller Software and Logic 1–6

Optional Equipment 1–6
- Serial Upload/Download Cable (RS-232) 1–6
- Remote Keyswitch and RS-232 Port Assembly (1200/1200e/1400e Terminals Only) 1–7
- Printer 1–7
- Uploads and Downloads over DH+ 1–7
- Uploads and Downloads over ControlNet 1–7
- PCMCIA Memory Card 1–7

Planning Applications — Chapter 2

Steps for Building an Application 2–1
Planning Steps 2–2
Planning for Safety 2–2
- Programming the Controller for Safety 2–3

Tips for Optimizing the Application 2–4
Using Worksheets to Help Plan Your Application 2–5
Step 1: Document the Process 2–5
Step 2: Design Application Screens and Objects 2–5
- Tips for Planning Application Screen Designs 2–6
  - Consistency 2–6
  - Clarity 2–6
  - Usability 2–6
- Use a Screen Template to Ensure Consistency 2–7
- Using Worksheets to Design Screens 2–8
  - Screen Worksheet for PanelView 1200e Terminals 2–8
  - Screen Worksheets for PanelView 1000e or 1400e Terminals . 2–10

Step 3: Plan Communications with the PLC 2–12
Planning Remote I/0 Communications 2–12
- Identifying Inputs and Outputs 2–12
  - Understanding Addressing 2–12
  - Understanding Rack Assignments 2–13
  - Understanding Module Groups 2–14
  - Understanding Shared Racks 2–15
  - Understanding Full and Listen Only Access 2–16

Planning Rack Assignments 2–16
Discrete I/O Addresses 2–17
Block Transfer File Addresses 2–17
Determining Rack Size and Block Transfer File Requirements . 2–18
Reducing PLC Addressing Requirements 2–18
Planning Addresses 2–19
Using Worksheets to Plan Addresses 2–19
The Remote I/O Tags Worksheet 2–21
Recording Object Information 2–21
Recording Tag and Expression Information 2–22
Recording the Address 2–27
The Discrete I/O Usage Worksheet 2–28
The Block Transfer File I/O Usage Worksheet 2–29
The Rack Assignments Worksheet 2–31
Recording the PLC/Scanner Type 2–31
Selecting the Baud Rate 2–32
Assigning Racks 2–32
The Block Transfer File Worksheet 2–33
Assigning Pass-Through 2–34
Carrying Out Your Plan 2–34
Planning DH+ or ControlNet Communications 2–35
Using Worksheets to Plan Communications 2–35
The Communications Worksheet 2–36
Defining Nodes 2–37
Recording Node Information 2–37
DH+ Addressing 2–38
DH+ Networks Bridged by a 1785-KA 2–39
Multiple DH+ Networks Connected by a Pyramid Integrator . . . 2–40
Defining Scan Classes 2–42
Guidelines for Configuring Scan Classes 2–42
Recording Scan Class Information 2–43
Planning Addresses 2–43
Guidelines for Organizing Tags to Optimize DH+ Network Performance 2–43
Recording Object Information 2–45
Recording Tag and Expression Information 2–46
Recording the Address 2–50
Carrying Out Your Plan 2–50

Working with Applications — Chapter 3

Creating a New Application 3–1
Opening Applications 3–3
Converting Applications Created in PanelBuilder 1200 or PanelBuilder for DOS 3–4
The Conversion Process 3–5
Specifying Target Firmware Version 3–6
Saving Applications 3–7
Saving an Application in Another File Format 3–8
Copying Applications 3–10
Renaming Applications 3–10
Validating Applications 3–10
Determining the Application Size 3–11
Creating Application Reports 3–12
Report On Options 3–13
Report On Screens Options 3–14
Print Setup 3–15
Solving Printing Problems 3–15
Saving the Report as an ASCII Text File 3–16
Closing Applications 3–17
Deleting Applications 3–17

Working with Editors — Chapter 4

Dialog Box Editors 4–1
Form/Spreadsheet Editors 4–3
Using the Spreadsheet and Form 4–3
Moving between the Spreadsheet and the Form 4–4
Using Multiple Editors 4–5
Working in the Spreadsheet 4–6
Using the Undo and Redo Commands (Messages Only) 4–7
Selecting Cells, Rows and Columns 4–7
Cutting or Copying and Pasting Selections (Messages Only) 4–9
Copying and Pasting into Differently Sized Areas 4–9
Sorting Entries in the Spreadsheet 4–10
Sorting Portions of the Spreadsheet 4–10
Duplicating Rows 4–11
Filling Portions of the Spreadsheet (Messages Only) 4–11
Inserting a Row in the Spreadsheet 4–12
Deleting Data in the Spreadsheet 4–12
Editing Selections (Messages Only) 4–12
Changing Column Width and Row Height 4–13

Defining Communications

Chapter 5

What Are Communications? 5–1
Defining Communications for a Remote I/O Application 5–2
Communication Setup 5–2
Setting up Terminal Communications 5–3
Setting the Baud Rate 5–3
Setting Up Network File Transfers 5–4
Specifying the Pass-Through Network Type 5–4
Specifying the PanelView Location Routing Address 5–5
Defining the Network File Transfer Controls 5–7
Assigning Device Type and Racks for the PanelView Terminal 5–8
Specifying the Device Type 5–9
Assigning Racks 5–9
Setting Up Block Transfer Files 5–10
Creating a Block Transfer File 5–12
Defining the Pass-Through File Control Byte 5–13
Changing Rack or Block Transfer File Assignments 5–14
Defining Communications for a Data Highway Plus or ControlNet Application 5–15
Communication Setup 5–15
Setting Up Terminal Communications 5–16
Specifying the Network Type 5–17
Setting the Terminal Network Address 5–17
Setting Up Network File Transfers 5–18
Specifying the PanelView Location Routing Address 5–18
Defining the Network File Transfer Controls 5–19
Configuring Nodes 5–21
Unsolicited Messages (DH+) 5–23
Configuring Scan Classes 5–24
Changing the Network Type 5–25
Converting from Remote I/O to DH+ 5–26
Converting from Remote I/O to ControlNet 5–28
Converting from DH+ to Remote I/O 5–30
Converting from DH+ to ControlNet 5–31
Converting from ControlNet to Remote I/O 5–33
Converting from ControlNet to DH+ 5–34

Defining Tags

Chapter 6

What Are Tags? 6–1
Tag Types 6–2
  Read and Write Tags 6–2
Data Sources 6–2
  Device 6–3
  Remote I/O 6–3
Organizing Tags 6–3
  Naming Tags 6–4
  Using Folders to Group Tags 6–4
The Tag Database Editor 6–4
  Opening the Tag Database Editor 6–5
  Using the Editor 6–5
Using the Form 6–6
  Using the Buttons in the Form 6–6
Using the Query Box 6–6
Using the Folder Hierarchy 6–7
  Creating a Folder 6–7
  Nesting Folders 6–8
  Opening a Folder 6–8
  Adding Tags to a Folder 6–9
  Duplicating a Folder 6–9
  Deleting a Folder 6–9
Using the Spreadsheet Viewer 6–10
  Moving Through the Spreadsheet 6–10
  Resizing Columns and Rows 6–10
  Adding a Tag 6–10
  Duplicating a Tag 6–11
  Editing a Tag 6–11
  Renaming a Tag 6–11
  Deleting a Tag 6–11
Working in Multiple Editors 6–12
Configuring Tag Type 6–13
  Configuring an Analog Tag 6–13
  Configuring a Digital Tag 6–16
  Configuring a String Tag 6–17
  Configuring a Block Tag 6–18
Specifying a Data Source 6–20
  Device Data Source 6–20
  Remote I/O Data Source 6–21
  Relaxed Validation 6–21
  Specifying Device as the Data Source 6–21
  Specifying Remote I/O as the Data Source 6–23
Other Methods of Creating Tags 6–23
  Creating Tags as Needed 6–23

Using the Tag Usage Viewer 6–25
Filtering Data 6–26
Searching for Text 6–28
Renaming a Tag 6–30
Private and Shared Tag Databases 6–30
The Private Database 6–30
The Shared Database 6–31
Using Shared Tag Databases 6–32
Converting a Private Database to Shared 6–33
Selecting a Shared Database 6–34
Repairing a Shared Database 6–35
Tag Database Import and Export 6–35
CSV File Column Ordering 6–37
Creating a New Tag Database 6–38

Creating Expressions

Chapter 7

What Are Expressions? 7–1
Types of Expressions 7–2
How Expressions Work 7–2
Expression Data Types 7–3
Tag Data Types in Expressions 7–4
Expression Language 7–4
Types of Operations Expressions Can Perform 7–4
Order of Precedence 7–4
Other Expression Components 7–5
Local Variables 7–6
Evaluation of Local Variables 7–7
Expression Maximums 7–8
Arithmetic Operations 7–9
Comparison Operations 7–11
Construct Operations 7–12
Logical Operations 7–16
Expression Return Values 7–19
Expression Compilation and Regeneration 7–20
Using the Expression Editor 7–21
About the Expression Editor Window 7–21
Entering and Formatting Expressions 7–22
Checking Syntax 7–23
Saving Expressions 7–24
Assigning Write Expressions 7–24
User Entry Character 7–24
Validating Expressions 7–27
More Expression Examples 7–28

Creating Screens

Chapter 8

Creating a New Application Screen 8–1
Opening Screens 8–3
Copying Application Screens within the Same Application 8–3
Copying Application Screen Contents into Another Application 8–4
Renaming Screens 8–5
Updating Screens 8–6
Closing Screens 8–7
Deleting Screens 8–7
Setting a New Screen Background Color 8–7
Assigning Security Access to Screens 8–8
Validating Screens 8–9
Viewing the Tags Used in a Screen 8–10
Working with Grids 8–11
- Using the Screen Grid 8–11
- Using the Touch Cell Grid (for PanelView 1200e Applications Only) 8–13

Displaying Function Keys (Keypad Applications Only) 8–14
Assigning the Application Startup Screen 8–15

Creating Objects

Chapter 9

What Is an Object? 9–1
Types of Objects 9–1
- Dynamic and Static Objects 9–2
- Display and Input Objects 9–2
- About Retentive Objects 9–3
- Initial Tag Values and Initial State Values 9–3
- Safety Considerations for Retentive Objects 9–4
- Retentive Objects and Tags 9–4

Objects Available in PanelBuilder 9–7
- ASCII Display 9–7
- ASCII Input 9–7
- Bar Graphs 9–7
- Control Selectors 9–7
- Goto Configure Mode 9–8
- Graphic Images 9–8
- Indicators 9–8
- Local Message Display 9–9
- Numerics 9–9
- Push Buttons 9–10
- Scales 9–11
- Screen Print Button 9–11
- Screen Selectors 9–11
- Scrolling List 9–11
- Shapes (Drawing Objects) 9–12

Symbols 9–12
Text 9–12
Time and Date 9–13
Trends 9–13
Working with Objects 9–13
Using the Toolbox and Color Palette 9–13
Selecting Objects 9–14
Selection Handles 9–15
Selection Methods 9–15
Deselecting Objects 9–18
Zooming In and Out 9–18
Using the Undo and Redo Commands 9–19
Object Tools and Commands 9–20
Creating Dynamic and Static Objects 9–22
Creating Text 9–24
Editing Text 9–25
Creating Extended Characters 9–25
Creating Symbols 9–26
Exporting Symbols 9–28
Creating Graphic Images 9–29
Exporting Graphic Images 9–31
Creating Drawing Objects 9–33
Adding Labels 9–34
Editing Objects 9–35
Cutting, Copying, and Pasting 9–36
Copying Objects with Controls 9–36
Copying Graphic Images 9–36
Deleting Objects 9–37
Duplicating Objects 9–37
Sizing Objects 9–38
Moving Objects 9–39
Converting Static Objects to Wallpaper 9–40
Arranging Objects 9–40
Grouping and Ungrouping Objects 9–41
Editing Overlapping Objects 9–42
Aligning Objects 9–43
Spacing Objects 9–45
Flipping Objects 9–45
Arranging Labels 9–46
Changing Object Appearance 9–46
Setting the Object's Foreground and Background Colors 9–47
How Colors Appear at Runtime 9–47
Setting an Object's Fill Type 9–48
Changing the Border and Line Styles 9–48
Making Objects Blink 9–49

Changing Text Appearance 9–50
Changing Touch Margin Width or Height (Touch Screen Only) . . . 9–50
Working with Multi-Component Objects 9–51
Setting Object Defaults 9–52
Configuring Objects 9–53
Assigning Function Keys (Keypad Applications Only) 9–54
Configuring Object States 9–55
Configuring Number of States 9–56
Assigning Values 9–57
Error States 9–58
Configuring List Object States 9–59
Adding Text to List Object States 9–60
Configuring Initial States 9–60
Assigning Tags and Expressions to Objects 9–61
Viewing Assigned Tags 9–63

Creating Messages **Chapter 10**

What Are Messages? 10–1
Kinds of Messages 10–1
How Messages Are Triggered 10–2
Creating and Editing Messages 10–3
Guidelines for Creating and Editing Messages 10–3
Using Formatting Codes in Your Messages 10–4
Version 2 or Later Applications 10–5
Creating a New Message 10–6
Adding Audio, Print, Display, Alarm Relay, and Message to PLC Options to Alarm Messages 10–8
Importing Messages 10–9
Importing Messages Using the Clipboard 10–9
Importing Existing Message Files 10–10
Guidelines for Importing Message Files 10–10
Format of the Message File 10–11
Exporting Messages 10–13
Validating Messages 10–13
Updating Messages 10–14
Closing the Messages Editor 10–15

Configuring Alarms **Chapter 11**

What Are Alarms? 11–1
How Are Alarms Triggered? 11–2
Alarm Indication Options 11–2
How Are Alarms Displayed? 11–3
How Are Alarms Acknowledged? 11–5
Configuring Alarms 11–5
Configuring Alarms' Global Controls 11–6

Alarm Trigger (Bit or Word?) 11–6
Alarm Acknowledge to PLC 11–8
Control 11–9
Alarm Triggers 11–9
Acknowledge to PLC and Acknowledge Control 11–15
Remote Acknowledge 11–19
Silence Alarms to PLC 11–22
PLC Controlled Silence Alarms 11–22
PLC Controlled Audio 11–22
PLC Controlled Relay 11–23
Message to PLC 11–24
Full Handshaking 11–24
Partial Handshaking 11–26
No Handshaking 11–27
Message Control 11–28
Message Handshake 11–28
About the Alarm Message Window 11–29
About Background Alarms 11–29
Using the Alarm Message Window in the Terminal 11–30
Single-line Alarm Message Window 11–30
Multi-line Alarm Message Window 11–31
Using Buttons in the Multi-line Alarm Message Window 11–32
Alarm Message Panel 11–33
Configuring the Alarm Message Window 11–35
About the Alarm Status Screen 11–39
Working with the Alarm Status Screen in the Terminal 11–40
Calling up the Alarm Status Screen from the Terminal 11–41
Viewing the Alarm Messages 11–41
Working in Display Active Alarms Mode 11–42
Working in Display All Alarms Mode 11–43
Working in Past Alarms Mode 11–43
About Alarm Quantity and Accumulated Time 11–44
Using the Alarm Status Screen Buttons 11–45
Resetting the Alarm Status Screen's QTY/TIME 11–46
What Happens when the Operator Resets the QTY/TIME? . . . 11–46
What Happens when the Terminal is Off Line? 11–47
Configuring the Alarm Status Screen 11–48
About the Alarm History Screen 11–50
Working with the Alarm History Screen in the Terminal 11–50
Calling up the Alarm History Screen from the Terminal 11–50
Using Buttons on the Alarm History Screen 11–51
Configuring the Alarm History Screen 11–52

Configuring Terminal Setup Options

Chapter 12

About Terminal Setup Options 12–1
General Parameters 12–2
The Application Startup Screen 12–2
Changing the Application Type 12–2
Describing the Application 12–3
Target Firmware Version 12–4
Specifying Target Firmware Version 12–4
Timing Parameters 12–5
Setting the Push Button Hold Time 12–6
Setting the Remote Alarm Parameters 12–7
Setting the Enter Key Parameters 12–8
Setting the Information Message Window Acknowledge Hold Time 12–8
Setting the Auto-Repeat Functions 12–8
Object Setup 12–9

Configuring PLC I/O Control Options

Chapter 13

Configuring PLC I/O Control Options 13–1
PLC Controlled Options 13–3
PLC Controlled Screen Number 13–3
PLC Controlled Screen Print 13–4
About Screen Prints from the Terminal 13–4
PLC Controlled Date & Time Control 13–4
Individual PLC Controlled Date & Time Controls 13–5
Notify PLC Options 13–6
Screen Number to PLC 13–6
Screen Print Active to PLC 13–6
Date & Time to PLC 13–6

Working with the Information Message Window

Chapter 14

About the Information Message Window 14–1
Using the Information Message Window in the Terminal 14–2
Configuring the Information Message Window 14–3

Transferring Applications

Chapter 15

About Transferring Applications 15–1

Sample Worksheets — Appendix A

The Extended Character Set — Appendix B

Addressing Syntax — Appendix C

PLC Addressing Syntax C–1
- PLC-2 . . . . . C–1
- PLC-3 Data Table Sections C–2
- PLC-3 Timers and Counters C–2
- PLC-5 Data Files C–3
- PLC-5 Binary Files: Optional Syntax C–3
- PLC-5 I/O and Status Files C–4
- PLC-5 Timers, Counters, and Control Files C–4
- PLC-5 (Enhanced) I/O and Status Files C–5
- PLC-5 (Enhanced) Binary, Integer, BCD, ASCII, Float, and String Files C–5
- PLC-5 (Enhanced) Binary Files: Optional Syntax C–6
- PLC-5 (Enhanced) Timer, Counter, Control, SFC Status, Message, PID, Block Transfer, and Token Data Files C–6

SLC Addressing Syntax C–7
- SLC 5 (Enhanced) I/O Files C–7
- SLC 5 (Enhanced) Status Files C–8
- SLC 5 (Enhanced) Binary, Integer, Float, ASCII, and String Files C–8
- SLC 5 (Enhanced) Binary Files: Optional Syntax C–9
- SLC 5 (Enhanced) Timer, Counter, and Control Files C–9
- SLC 5 (Enhanced) Timer, Counter, and Control Files Bit Member Addressing: Optional Syntax C–10
- SLC 5 (Enhanced) Timer, Counter, and Control Files Bit Member Addressing by Bit Address: Optional Syntax C–10
- SLC 5 (Enhanced) Timer, Counter, and Control Files Analog Member Addressing: Optional Syntax C–11

Mnemonic Tables C–12
- Timer Mnemonics C–12
- Counter Mnemonics C–12
- Control Mnemonics C–12
- PID Mnemonics C–13
- Message Mnemonics C–14
- Block Transfer Mnemonics C–14
- Token Data Mnemonics C–15
- SFC Status Mnemonics C–15

Remote I/O C–16
- Discrete PLC Address C–16
- Block Transfer Address C–17
- Discrete SLC Address C–17

Troubleshooting

Appendix D

PanelBuilder Problems D–1
PanelBuilder Validation Errors D–3

Version Control

Appendix E

What Is Version Control? E–1
Version Control and Application Validation E–1
Target Firmware Version vs. Actual Firmware Version E–1
Version 1.03 Firmware Features E–2
Version 2 Firmware Features E–2
Version 3 Firmware Features E–3
Version 4 Firmware Features E–3
Version Control and Downloading Applications E–3
Version Control at the PanelView Terminal E–4

Reserved Keywords

Appendix F

Index

Preface

Welcome to PanelBuilder 1400e Configuration Software for Windows

Welcome to Allen-Bradley's PanelBuilder™ 1400e Configuration Software for Windows, Version 4. With this software you can create applications in the Microsoft® Windows® 3.1 (or later) operating system, in Windows 95, and in Windows NT 4.0. You can use the PanelBuilder applications in PanelView™ 1000e, 1200e, or 1400e terminals, or PanelView 1200 Series F and later terminals that have been enhanced to -MC catalog numbers.

PanelBuilder 1400e, Version 4 provides expanded hardware support to allow you to create applications for a wider variety of environments, terminals, and networks:

Windows NT Compatibility

PanelBuilder 1400e Version 4 has been enhanced to run on a 32-bit operating system, specifically Windows NT 4.0. However, PanelBuilder 1400e, a 16-bit application, may still run on the 16-bit platforms (Windows 3.1 or later, and Windows 95).

ControlNet Station Addressing

With PanelBuilder Version 4, you can now assign the station addressing on ControlNet from 1 to 99. Previously, you could only assign addresses above 63 through the network setup.

Ethernet Pass-Through

Network users now have the capabilities of downloading or uploading PanelBuilder 1400e applications over an Ethernet® network to a PanelView on a Remote I/O network. With the installation of an Ethernet network card in your PC, you can develop your PanelBuilder applications program on an Ethernet network, and download or upload to your PanelView terminals operating on a Remote I/O network.

The Ethernet network card utilizes the PLC 5E as a pass-through device, enabling your application to download to the PanelView terminal.

CD-ROM Installation

PanelBuilder Version 4 can now be installed from a CD-ROM. This not only reduces the number of disks required, but simplifies the installation of PanelBuilder 1400e. However, for those who want disk sets, they are available upon request. Send in the enclosed Response Card to receive the required package of disks.

RSLinx Communication Drivers

PanelBuilder 1400e Version 4 includes RSLinx communication drivers for application file transfers on Windows NT and Windows 95. INTERCHANGE and WINLinx are still available for Windows 3.1 or later, and Windows 95, if desired.

Terminology

The term *PanelBuilder* refers to PanelBuilder 1400e Configuration Software for Windows. Where confusion may arise between the current and previous versions of the software, the current release of the software is called “PanelBuilder 1400e, Version 4.”

Similarly, the terms *terminal* and *PanelView terminal* refer to a PanelView 1000e, 1200e, or 1400e terminal or an enhanced PanelView Series F or G terminal. Where confusion may arise between the 1000e, 1200e, and 1400e terminals and previous revisions, specific series and revision names are used.

The terms *PLC* and *programmable controllers* refer to the Allen-Bradley line of PLC® Programmable Controllers.

The term *control* is a generic term that refers to the PLC addresses that dynamic objects write to or read from. Some controls use tags only, while others can use tags or expressions. All dynamic objects use one or more controls. In this manual, the configuration table for each object lists the controls for the object, and specifies which controls use tags only and which can use expressions. For more information about expresssions see Chapter 7, *Creating Expressions*, in this manual.

Registering Your Copy of PanelBuilder 1400e

To register your software, mail the registration card from the front of this manual to this address:

Rockwell Software
Software Services
6680 Beta Drive
Mayfield Village, Ohio 44143

or fax the card to 1-440-646-7701.

Available Documentation

Your PanelBuilder software comes with several types of documentation to meet your different needs:

- *Getting Started with PanelBuilder 1400e Configuration Software for Windows* (Publication Number 2711E-818) guides you through setting up PanelBuilder 1400e and introduces you to the basics. It includes a tutorial to give you hands-on experience in working with a PanelBuilder application.
- The *PanelBuilder 1400e Configuration Software for Windows User Manual* (Publication Number 2711E-819) explains PanelBuilder in more detail, and provides step-by-step instructions for planning, creating, and working with applications.
- The *PanelBuilder 1400e Configuration Software for Windows Reference Manual* (Publication Number 2711E-820) provides detailed reference information for application screen objects.
- The *PanelView 1200/1400e Transfer Utility User Manual* (Publication Number 2711E-6.8) provides detailed instructions for transferring application files using the Transfer Utility, Version 4, that comes with PanelBuilder 1400e, Version 4.
- Context-sensitive online help provides a quick reference for procedures or commands you need explained, or problems you may encounter. To get help, press F1 or choose the Help button if you're in a window or dialog box.
- The *PanelBuilder 1400e Readme* file is a Microsoft Windows Notepad file that is copied to your hard disk when you install PanelBuilder. The *Readme* file informs you of any software changes made after the manuals were printed.
- The *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual* (Publication Number 2711E-821) describes how to install, configure, maintain, and troubleshoot the PanelView terminal.
- The *PanelBuilder 1400e Configuration Software for Windows Modbus User Manual*, Publication Number 2711E-6.12, describes how to create PanelBuilder applications for the Modbus communications network. This manual is supplied as part of the optional Modbus Communications Kit, Catalog Number 2711E-UMOD.

What's in the *User Manual?*

The *User Manual* builds on the information presented in *Getting Started.* The first part of the *User Manual* provides detailed information about planning your application, as well as information about how to use PanelBuilder. The remainder of the manual provides step-by-step instructions for building, configuring, and transferring applications.

Preface

An overview of this manual and lists of related publications, documentation conventions, and prerequisites.

Chapter 1, Introduction to PanelBuilder

An overview of PanelBuilder, its editors, and required and optional equipment.

Chapter 2, Planning Applications

An outline of what you need to do before you create an application, including an overview of the steps for building an application and how to plan for safety and optimal runtime performance. This chapter also explains in detail the planning steps of documenting the process your application will control or monitor, designing your screens, and planning communications.

Chapter 3, Working with Applications

Instructions for working with applications, such as creating, opening, saving, and closing applications.

Chapter 4, Working with Editors

Instructions for working with dialog boxes and the form/spreadsheet style editors used to configure an application.

Chapter 5, Defining Communications

Instructions for defining PLC communications for Remote I/O applications, including selecting the controller and baud rate, assigning racks and block transfer files, and setting pass-through communication parameters. For DH+ and ControlNet applications, the instructions include selecting the controller and baud rate, and defining nodes and scan classes.

Chapter 6, Defining Tags

Information about using the Tag Database editor to define tags.

Chapter 7, Creating Expressions

Information about using the Expression editor to define expressions.

Chapter 8, Creating Screens

Instructions for working with application screens and using some of the tools.

Chapter 9, Creating Objects

An overview of the different types of application screen objects you can create in PanelBuilder, and step-by-step instructions for creating, editing, and configuring objects.

Chapter 10, Creating Messages

Information about the different types of messages and how they work, and instructions for creating, editing, and importing messages.

Chapter 11, Configuring Alarms

General discussion of how alarms are reported on a PanelView terminal, and details about how to configure the various alarm reporting options.

Chapter 12, Configuring Terminal Setup Options

Information about configuring options that determine how the application appears on the terminal.

Chapter 13, Configuring PLC I/O Control Options

Information about configuring options that determine how the application interacts with the PLC.

Chapter 14, Working with the Information Message Window

Information about configuring the Information Message Window object to display messages on the terminal's screen.

Chapter 15, Transferring Applications

Information about transferring applications between the development computer and PanelView terminal using the Upload/Download cable, RIO Pass-Through, direct network downloads and uploads, or the PCMCIA memory card.

Who Should Read This Manual?

This manual is for users who have a basic knowledge of PanelBuilder, and a good knowledge of Microsoft Windows. Users who are unfamiliar with PanelBuilder should first read *Getting Started*.

Users who are not familiar with Microsoft Windows should read their *Microsoft Windows User's Guide* (for users of Windows 3.1), *Introducing Microsoft Windows 95* (for users of Windows 95), or *Introducing Microsoft Windows NT Workstation* (for users of Windows NT).

Related Publications

The following table lists Allen-Bradley remote I/O scanner module user manuals:

Publication	Number
1772-SD/SD2 Remote I/O Scanner/Distribution Panel	1772-2.18
1775-S4A I/O Scanner-Programmer Interface Module User's Manual	1775-6.5.1
1775-S5, 1775-SR5 I/O Scanner-Communication Adapter Module User's Manual	1775-6.5.5
5150-RS PI Start-up and Integration Manual	5000-6.5.1
6008-SI IBM® PC I/O Scanner User's Manual	6008-6.5.3
6008-SV VME I/O Scanner User's Manual	6008-6.5.2
6008-SQ Q-Bus I/O Scanner Utility Software User's Manual	6008-6.4.1
1771-SN Sub I/O Scanner Module Data Sheet	1771-2.91
1747-SN RIO Scanner User's Manual	1747-NM005

The following table lists Allen-Bradley programmable controller user manuals:

Publication	Number
1772-LP2 PLC-2/20 Programming and Operations Manual	1772-6.8.1
1772-LP3 PLC-2/30 Controller Programming and Operations Manual	1772-6.8.3
PLC-3 Family Controller Programming Reference Manual	1775-6.4.1
PLC-5 Family Programmable Controllers Hardware Installation Manual	1785-6.6.1
PLC-5 Programming Software	6200-6.4.7
5250-LP1, LP2 PLC-5/250 Programming Manual	5000-6.4.8
SLC 500 Family of Programmable Controllers Advanced Programming Software User Manual	1747-NM002
ControlNet PLC-5 Programmable Controllers User Manual	1785-6.5.14

Conventions Used

Information is provided in a consistent way throughout all the PanelBuilder 1400e user documentation. The documentation uses these print conventions, mouse conventions, selection conventions, key conventions, and command conventions:

Print Conventions

Applications can be created that communicate with PLCs over Remote I/O, DH+, or ControlNet. When instructions apply specifically to one type of application or the other, this is noted in the margin:

RIO
- means instructions for a Remote I/O application only

DH+
- means instructions for a Data Highway Plus (DH+) application only

ControlNet
- means instructions for a ControlNet application only

DH+ and ControlNet
- means instructions for DH+ and ControlNet applications but not Remote I/O applications

All network types
- means instructions for all applications, regardless of communication network type

Mouse Conventions

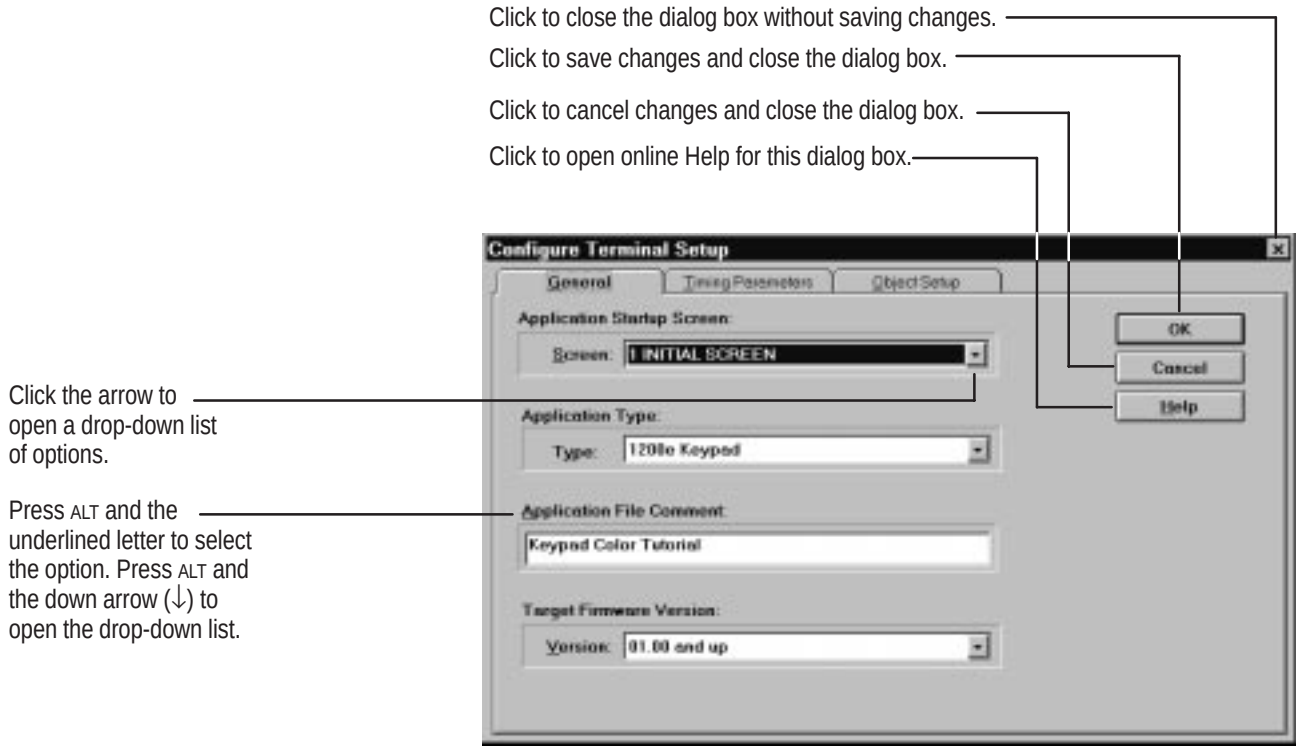
You can use a mouse with one or two buttons. This manual assumes that if you have a multiple-button mouse, the left mouse button is configured as the primary mouse button. Procedures that require you to click a secondary button refer to it as the right mouse button.

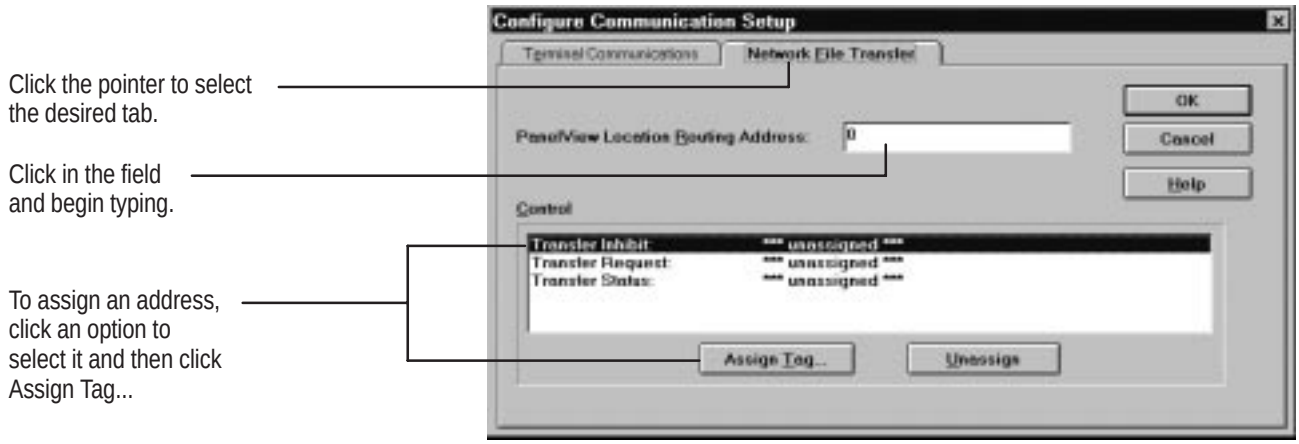
Selection Conventions

This word or phrase	Means
Choose	Execute a command from a menu or from a button in a dialog box or Help window. Choose also means to double-click an icon.
Choose OK	Either click the OK button with the mouse or press Enter on the keyboard.
Select	Either highlight the object or piece of text you want your next action to affect, or select a specific dialog box option.
Click	Position the mouse pointer on the object, area, or field, and click the left button once.
Double-click	Position the mouse pointer on the object, area, or field, and click the left button twice quickly.
Right-click	Click the right mouse button.

Selection Conventions in Dialog Boxes

Dialog boxes contain various features that require different selection conventions. In the following Windows NT example, the Configure Communication Setup dialog box is used to illustrate common dialog box features and selection conventions. All screen captures in this manual reflect the Windows NT user interface. Significant differences are highlighted wherever they occur.


Click to close the dialog box without saving changes.
Click to save changes and close the dialog box.
Click to cancel changes and close the dialog box.
Click to open online Help for this dialog box.
Click the arrow to open a drop-down list of options.
Press ALT and the underlined letter to select the option. Press ALT and the down arrow (↓) to open the drop-down list.
Configure Terminal Setup
General
Timing Parameters
Object Setup
Application Startup Screen:
Screen:
1 INITIAL SCREEN
Application Type:
Type:
1200e Keypad
Application File Comment:
Keypad Color Tutorial
Target Firmware Version:
Version:
01.00 and up
OK
Cancel
Help



The following shortcut keys allow you to complete dialog boxes faster:

This key or key combination	Does this
TAB	Moves the cursor to the next field, option, or command button.
SHIFT+TAB	Moves the cursor to the previous field, option, or command button.
ALT+underlined letter	Selects an option.
ALT+↓	Displays a drop-down list.
SPACEBAR	Turns check boxes on or off.

Key Conventions

This key combination	Means
KEY1+KEY2	Press and hold the first key while you press the second key. For example, press "CTRL+A" means press the CTRL key, and while pressing it, press the A key. Then release both keys.
SHIFT+click	Press and hold the SHIFT key while you click an object with the pointer.

Command Conventions

There are different ways to carry out commands in PanelBuilder 1400e. Often, you can execute the same command in three ways by:

- choosing a menu command
- choosing an icon from the toolbar
- using a key combination

For example, use any of the following methods to open an application:

- choose Open Application from the File menu
- choose from the toolbar
- press CTRL+O

Note: Instructions in this manual do not always outline all three methods.

Before You Begin

You need the following before you begin:

- a personal computer with at least a 486, 25-MHz microprocessor; at least 8 MB Random Access Memory (RAM) for Windows 3.1/95 (although 16 MB RAM is recommended for Windows 95); at least 32 MB RAM for Windows NT; and a SVGA monitor with 256 colors (recommended). For users working with imported .dxf files, at least 16 MB RAM is required.
- If you want to resize graphic images in PanelBuilder, set your display adapter to 65,536 colors
- Microsoft Windows 3.1 (or later), Windows 95, or Windows NT 4.0 operating system
- the applicable programmable controller documentation
- *Getting Started with PanelBuilder 1400e Configuration Software for Windows*

Technical Support Services

If you have questions about PanelBuilder, please consult the manuals or the online Help first. If you can't find the answer, take advantage of our Technical Support Fax Back system, available 24 hours a day, 7 days a week at 1-440-646-5436, or browse through our technical support document library at **http://www.ab.com/mem/prodserv/services/technotes/techmain.html** on the World Wide Web.

Alternatively, contact:

Allen-Bradley
Technical Support
1 Allen Bradley Drive
Mayfield Heights, Ohio 44124-6118

or call 1-440-646-6800 or fax 1-440-646-6890 for technical support between 8 AM and 5 PM (EST), Monday to Friday.

Please have the serial number for your software ready when you call, or include it on your fax. You can find this number in three places:

- on the Software Registration card that was shipped with your software
- on the screen that appears when you start PanelBuilder
- in the main Help menu, when you choose "About"

Introduction to PanelBuilder

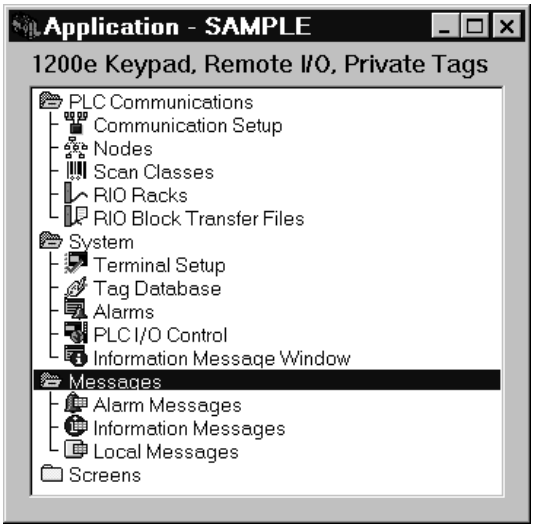
PanelBuilder 1400e Configuration Software for Windows is a configuration environment based on Microsoft Windows. Use PanelBuilder 1400e to develop operator interface applications for PanelView 1000e, 1200e, and 1400e operator terminals, or PanelView 1200 Enhanced Series F or later terminals that have been enhanced to –MC catalog numbers.

The Windows platform offers many advantages, including a superior graphical interface, the ability to work within several applications at once, and easy information transfer between applications.

PanelBuilder is easy to learn and use, and helps you develop complex applications quickly and efficiently.

PanelBuilder Components

PanelBuilder contains the following editors and dialog boxes to help you build and configure your applications:



- Communications Setup dialog box for configuring communications (baud rate and network type) between the terminal and programmable controller, and for defining control for downloading applications over the communications network
- Node and Scan Class editors for specifying programmable controllers and the scan rates at which the application will communicate

RIO

- RIO Racks and RIO Block Transfer Files editors for configuring the racks, block transfer file definitions, and pass-through control byte for Remote I/O application file transfers
- Terminal Setup dialog box for specifying the startup screen, application type, application comment, and setting timing parameters
- Tag Database editor for defining programmable controller addresses required by the application
- Alarms dialog box for configuring the alarm information that informs users about problems when the application is running
- PLC I/O Control dialog box for assigning tags to PLC Controlled and Notify PLC options
- Information Message Window dialog box for configuring the look and function of the Information Message window
- Three Message editors for creating and editing messages
- Screen editor for designing the application screens and configuring the application screen objects that allow the operator to monitor and control the process or machine

What Are You Building?

Use PanelBuilder to build applications that run on PanelView terminals. PanelBuilder applications help automate industrial processes, allowing an operator to monitor and control a single machine or an entire process through the terminal.

Version Compatibility

Applications created in PanelBuilder 1400e Configuration Software for Windows, Version 4, can run on:

- PanelView 1000e, 1200e, and 1400e operator terminals
- Series F and G PanelView 1200 terminals that have been enhanced to -MC catalog numbers with PanelView 1200e and 1400e Firmware, 2-MB SIMM module, and 4-MB PCMCIA memory card. For information about enhancing your Series F or G terminals, contact your Allen-Bradley representative.

You can edit any application created with a previous version of PanelBuilder (for example, an application created in Version 1 of PanelBuilder DOS). However, once an application is edited in PanelBuilder 1400e, you can save it only as a Version 4 PanelBuilder 1400e format file (.pvc), and you can only edit it using PanelBuilder Version 4. Depending on the features you used in the application, you can, however, download and run it on terminals running PanelView 1000e, 1200e and 1400e Firmware, Version 1 and later. Version compatibility issues are highlighted wherever they occur in this manual, and are explained in more detail in Appendix E, *Version Control.*

Note: The .pvd files are terminal files that are created automatically when an application is downloaded to a PanelView terminal. You can also create a .pvd file manually by saving a PanelBuilder application as a terminal file. Terminal (.pvd) files can be Version 1 or later, depending on the features used in the application.

Required Equipment

You need the following software and equipment to create PanelBuilder applications and run them on a PanelView terminal.

PanelBuilder Software

You need a copy of PanelBuilder 1400e Configuration Software for Windows, Version 3 or 4, installed on your personal computer. For PC requirements and installation instructions, refer to Chapter 2, *Introducing PanelBuilder 1400e,* in *PanelBuilder 1400e Configuration Software for Windows Getting Started Manual.*

Panelbuilder Transfer Utility

You need a copy of the PanelView 1200/1400e Transfer Utility, Version 3 or PanelView e FTU32 Version 4 installed on your personal computer if you want to transfer files between your computer and a PanelView terminal. For PC requirements and installation instructions, refer to the *PanelBuilder 1400e Transfer Utility User Manual*, Publication Number 2711E-ND7.

PanelView Terminal

You must have one or more terminals. There are fifteen types of terminals on which applications created or edited in PanelBuilder 1400e Configuration Software for Windows, Version 3 or 4, can run:

- PanelView 1000e Touch Screen Terminals
  (Catalog Nos. 2711E-T10C6, 2711E-T10C7, 2711E-T10C15)
- PanelView 1000e Keypad Terminals
  (Catalog Nos. 2711E-K10C6, 2711E-K10C7, 2711E-K10C15)
- PanelView 1200e Touch Screen Terminals (clip-mount and stud-mount)
  (Catalog Nos. 2711E-T12C6, 2711E-T12C4)
- PanelView 1200e Keypad Terminals
  (Catalog No. 2711E-K12C6)
  Stainless steel model: (Catalog No. K12C6L2)
- PanelView 1400e Touch Screen Terminals
  (Catalog Nos. 2711E-T14C6, 2711E-T14C7, 2711E-T14C15)
- PanelView 1400e Keypad Terminals
  (Catalog Nos. 2711E-K14C6, 2711E-K14C7, 2711E-K14C15)

- PanelView 1200 Color Keypad Terminal, Series F and G (Catalog No. 2711-KC1MC), when upgraded with the PanelView 1200 Enhancement Kit, Catalog Number 2711E-U1B12C, and running Version 1 or later firmware
- PanelView 1200 Color Touch Screen Terminal, Series F and G (Catalog Nos. 2711-TC1MC, 2711-TC4MC), when upgraded with the PanelView 1200 Enhancement Kit, Catalog Number 2711E-U1B12C, and running Version 1 or later firmware

Transferring or Installing the Application

To run an application, you must first transfer it from the development computer to the PanelView terminal using one of the following methods:

- Serial download using the RS-232 port and cable for a Remote I/O, DH+, or ControlNet application
- DH+/RIO Pass-Through download over DH+ for a Remote I/O application
- ControlNet/RIO Pass-Through download over ControlNet for a Remote I/O application
- Remote I/O/Ethernet Pass-Through download over an Ethernet for a Remote I/O application
- DH+ direct download over DH+ for a DH+ application
- ControlNet direct download for a ControlNet application
- PCMCIA memory card for a Remote I/O, DH+ or ControlNet application (Databook-compatible drive)

For more information about equipment necessary for transferring applications, see “Optional Equipment” in this chapter.

To transfer application files to or from PanelView terminals, the PanelView 1200/1400e Transfer Utility must be installed. This utility is included with your PanelBuilder distribution. It is also available separately as Catalog Number 2711E-ND7.

For information about installing the PanelView 1200/1400e Transfer Utility, and transferring application files, see the *PanelView 1400e Transfer Utility User Manual*, Catalog Number 2711E-6.8.

Programmable Controllers

PanelView terminals can be connected to a programmable controller as follows:

RIO

- PanelView terminals can be connected to any Allen-Bradley 1771 Remote I/O Link. Applicable host controllers include almost all Allen-Bradley Programmable Logic Controllers as well as certain IBM computers, VME Controllers, and the DEC Q-Bus interface.

  Newly-released Allen-Bradley programmable controllers (not listed) that support the 1771 Remote I/O link will support PanelView terminals.

  The following table lists the available programmable controller and scanner types.

Programmable Controller Type	Scanner Type	Rack Address Ranges
PLC-2	1772-SD/SD2	1 – 7 (Octal)
PLC-3	1775-S4A/B	0 – 76 (Octal)
PLC-3	1775-S5	0 – 76 (Octal)
PLC-3/10	1775-SR5	0 – 17 (Octal)
PLC-5/11	Built in	3 Only
PLC-5/15	Built in	1 – 3 (Octal)
PLC-5/20	Built in	1 – 3 (Octal)
PLC-5/20C	Built-in	1 – 3 (Octal)
PLC-5/25	Built in	1 – 7 (Octal)
PLC-5/30	Built in	1 – 7 (Octal)
PLC-5/40	Built in	1 – 17 (Octal)
PLC-5/40C	Built in	1 – 17 (Octal)
PLC-5/60	Built in	1 – 27 (Octal)
PLC-5/80	Built in	1 – 27 (Octal)
PLC-5/250	5150-RS	0 – 37 (Octal)
SLC 5/02, 5/03, 5/04	1747-SN Ser A	0 – 3 (Decimal)
SLC 5/02, 5/03, 5/04	1747-SN Ser B	0 – 3 (Decimal)
IBM PC	6008-SI	0 – 7 (Octal)
VME Bus	6008-SV	0 – 7 (Octal)
DEC Q-Bus	6008-SQ	0 – 7 (Octal)
Other	1771-SN Sub Scanner	1 – 7 (Octal)

For a description of each type of programmable controller, refer to Chapter 1, *Introducing PanelView Terminals*, in the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual*.

The terminal appears as one or more I/O racks to a programmable controller. It has the same configurability as—and more than—a standard I/O rack. Refer to the Allen-Bradley Programmable Controller and Remote I/O Scanner user's manuals for connection and remote I/O configuration limitations.

DH+

- You can connect the PanelView terminal to a DH+ network. The terminal can then communicate with all the controllers on that highway, and to controllers on a remote DH+ or Data Highway (DH) network via offlink addressing.

ControlNet

- You can connect the PanelView terminal to a ControlNet network. The terminal can then communicate with all the controllers on that network through unscheduled messaging.
- You can connect the PanelView terminal to a Modbus network. For more information, see the *PanelBuilder 1400e Modbus User Manual*, Publication Number 2711E-6.12, which is supplied as part of the optional Modbus Communications Kit, Catalog Number 2711E-UMOD.

Programmable Controller Software and Logic

You need the appropriate software to run the programmable controller. You also need to write a PLC program for the programmable controller so it can interface with the PanelBuilder application.

Optional Equipment

You can enhance your system with the following equipment:

Serial Upload/Download Cable (RS-232)

Use the Upload/Download cable to transfer applications between the development computer and terminal. If you have several terminals, or want a spare cable, you can order additional cables. For PanelView 1200, 1200e, and 1400e terminals, the cable is Catalog Number 2711-NC1. For PanelView 1000e terminals, the cable is Catalog Number 2711-NC13, 2711-NC14, or 2706-NC13.

Remote Keyswitch and RS-232 Port Assembly (1200/1200e/1400e Terminals Only)

All terminals have a Mode Select Keyswitch and RS-232 port on the back. If this is difficult to access, for PanelView 1200, 1200e, and 1400e terminals, you can install the Remote Keyswitch and RS-232 Port Assembly (Catalog No. 2711-NC2) in a more convenient location. Refer to the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual* for more information.

Printer

You can use any standard or serial printer supported by Microsoft Windows to print reports from the development computer. To print graphics screen reports from PanelBuilder, your printer must support graphics printing.

The PanelView terminal supports Epson® FX-80 compatible printers for printing screens and alarm reports.

Uploads and Downloads over DH+

PanelView terminals support DH+ Direct or RIO/DH+ Pass-Through file transfers. To upload or download your applications over DH+, you must connect your development computer to the Data Highway Plus network.

Uploads and Downloads over ControlNet

PanelView terminals support ControlNet Direct or RIO/ControlNet Pass-Through file transfers. To upload or download your applications over ControlNet, you must connect your development computer to the ControlNet network.

PCMCIA Memory Card

PanelView terminals provide Personal Computer Memory Card International Association (PCMCIA) support. You can use the PCMCIA memory card (Catalog No. 2711-NM11, 12, 13, or 14) to install firmware directly, without having to open the terminal, or to store application files.

For more information, see the *PanelView 1400e Serial Firmware Upgrade Utility* Readme file and online help.

Planning Applications

A well-designed application requires careful planning *before* you start building. To help with planning, this chapter covers:

- steps for building an application
- steps in planning an application, including documenting the process, designing screens, and identifying communications
- planning for safety
- tips for optimizing the application
- using worksheets

Steps for Building an Application

Creating an application is easier if you plan it. This chapter discusses planning steps and guidelines in detail. The later chapters of this manual provide instructions for creating and downloading the application. The chapters are organized in the order you use them to create your application. Follow these steps to build your application:

1. Plan your application, as described in this chapter.
2. Create the new application, as described in Chapter 3, *Working with Applications*. (Then read Chapter 4, *Working with Editors* for more information about how to work in PanelBuilder.)
   - **A.** Configure the communications for the application. This includes defining the nodes and scan classes for a DH+ or ControlNet application, and the racks and block transfer files for a Remote I/O application. See Chapter 5, *Defining Communications*.
   - **B.** Define all the tags and expressions you will need throughout the application. See Chapter 6, *Defining Tags* and Chapter 7, *Creating Expressions*.
   - **C.** Create the application screens and objects. See Chapter 8, *Creating Screens* and Chapter 9, *Creating Objects*.
   - **D.** Create the messages, if applicable. See Chapter 10, *Creating Messages*.
   - **E.** Configure the Alarm screens and controls, if applicable. See Chapter 11, *Configuring Alarms*.
   - **F.** Configure the options for how the application appears and functions at the PanelView terminal, and how it interacts with the PLC. See Chapter 12, *Configuring Terminal Setup Options* and Chapter 13, *Configuring PLC I/O Control Options*.

G. Save your application. See Chapter 3, *Working with Applications*.

3. Download the completed application to a PanelView terminal. See Chapter 15, *Transferring Applications*.
4. Test the application. See the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual* for information about testing the application on the PanelView terminal.

Planning Steps

Before you can begin creating an application, you need to follow these planning steps:

1. Document the process you'll be automating.
2. Design the application screens and objects.
3. Plan how the application will communicate with the PLC. This step includes several tasks:
   - Define nodes and scan classes for DH+ and ControlNet applications.
   - Identify all the PLC addresses needed for the screen objects, Alarm and Information windows, and PLC I/O Control options; then plan the tags and expressions that will link them to the PLC.
   - Define the rack assignments for discrete I/O and block transfer files for Remote I/O applications.

These steps are discussed in the remainder of this chapter.

Planning for Safety

At each stage of the planning process, keep safety considerations in mind.

ATTENTION: Do not rely on your PanelView terminal for emergency stops, or any other controls that are critical to the safety of workers or equipment. To ensure that the operation can be controlled safely, use separate, hard-wired operator interface devices that do not depend on solid state electronics. For more details about safety standards, refer to the inside cover of this manual.

The PanelView 1000e and 1400e touch screen terminals use analog-resistive touch screens. This allows input objects to be any size (minimum 40 by 40 pixels) and to be placed anywhere on the screen. The analog-resistive touch screen provides greater flexibility in screen design compared to PanelView 1200 and 1200e terminals. PanelView 1200 and 1200e terminals use a touch matrix, which requires that all input objects align with a touch grid.

ATTENTION: Do not press multiple touch screen locations at the same time. In the unlikely event that two or more touches occur simultaneously in the same horizontal or vertical line, an input object located half-way between the two touches could be activated. This is inherent in the analog-resistive touch screen technology used in PanelView 1000e and 1400e terminals.

Programming the Controller for Safety

The PanelView terminal will shut down its communications under the following conditions:

- terminal is switched to Configuration mode
- terminal loses AC power
- terminal has a major fault
- Remote I/O, DH+, or ControlNet link fails

In these situations, the PanelView terminal can no longer control or monitor programmable controller functions. In a Remote I/O application, the host controller sees the PanelView terminal as a faulted I/O rack. In a DH+ or ControlNet application, the terminal remains in the Active Node Table, but does not respond to communications, and cannot be accessed by other stations on the network. To clear the fault, switch the terminal to configure mode, then back to run mode, or cycle power to the terminal.

To ensure a safe recovery, program the controller to:

- return any machine or process to a safe state

RIO

- monitor the rack fault bit associated with the PanelView terminal's rack assignment. For information on the rack fault bit table, refer to your programmable controller documentation.

For additional information on programming the programmable controller for safety, refer to your programmable controller documentation.

Tips for Optimizing the Application

As you carry out the planning steps in the remainder of this chapter, keep these tips in mind. They can help you optimize your application's performance when it is running in the terminal.

- Convert all static objects—that is, text, lines, rectangles, ellipses, wedges, arcs, and graphic images—to wallpaper. This will speed up the screen refresh rate on the active screen.
- Use bitmaps sparingly.
- If you use the same bitmap more than once, ensure that each copy is the same size as the original. This means that the same bitmap is read each time it is used in the application. If you resize one or more of the copies, a new bitmap is saved for each size of the image, which can increase the size of your application significantly.
- Where bitmaps must be used, use monochrome rather than color bitmaps.
- Where possible, use a Symbol instead of a Multistate Indicator.
- Avoid creating two or more tags for the same PLC address. If you create two or more tags pointing to the same PLC address, the terminal will scan that same address location more than once.

For objects that have more than one state,

- Use state values that increment by one for each state. Do not configure more states than you need.
- If you need to minimize the size of the application file, do not assign names to objects. Names use one byte per character.

DH+ and ControlNet

- Avoid creating two or more node definitions for the same physical node.
- Set the background scan rate as slow as possible.
- Keep screens simple.
- Minimize or avoid object overlap.
- Try to ensure an object's State Values or Message Trigger Values can be accommodated by a byte or integer value, as opposed to a long integer.

For more information about configuring objects, see the *PanelBuilder 1400e Configuration Software for Windows Reference Manual.*

Using Worksheets to Help Plan Your Application

This manual comes with worksheets to guide you through planning your screens and identifying communications with the PLC. The remainder of this chapter includes instructions on using the worksheets as you plan. There are eleven sample worksheets in Appendix A of this manual. Make photocopies of the worksheets and use them as directed.

Step 1: Document the Process

Getting to know the process you'll be automating is the most important planning step. Try to find out as much as possible about the existing process and/or the one you plan to design.

To get information about the process, talk to operators, process engineers, and other experts who will be running the system. Ask questions such as:

- What specific tasks does the process need to perform?
- What controls do they need?
- Where should the controls go?
- What values and types of values are they working with?
- Should access to certain controls be restricted?
- What needs to be alarmed?

Next, find out which programmable controllers the process will be using, and how this will affect the application.

Tip: A questionnaire is a valuable tool to get this kind of information quickly.

When you are finished researching the process, make a comprehensive list of its individual operations and tasks.

Step 2: Design Application Screens and Objects

The second step in planning an application is designing the screens and objects. This section includes the following information:

- how to plan application screen designs
- how to plan for consistency across the application
- how to use worksheets as you plan screens and objects

Once you've completed the screens and decided which objects you'll use, you can plan the object addresses. Addressing is discussed later in this chapter.

Tips for Planning Application Screen Designs

Practising good screen design principles ensures that your application will be easy to use. When planning, keep these important principles in mind:

Consistency

- Be consistent with your use of symbols and color.
- Be consistent with button labels and button placement.

  When you design several screens, keep the same kinds of buttons in the same positions.

  For example, if you have a “Start” button in a certain position on one screen, don’t put a “Stop” button in the same position on the next screen.

Clarity

- Use easily-recognizable symbols.

  For example, use the conventional symbols provided as Graphic Images to draw objects such as tanks and valves.
- Don’t overload the screen with too much information.
- Use standard, clear terminology, and avoid abbreviations or acronyms that the user may not understand.
- Use colors with recognizable meanings.

  For example, in North America, red and green often signify stop and start; keep color meanings consistent by assigning red only to “Stop” buttons, and green only to “Start” buttons.

  Note: Because some people are color-blind, don’t rely on color alone to establish meaning.
- Use high contrast color combinations, such as yellow on blue.

Usability

- On keypad screens, place button labels near their associated function buttons.
- On touch screens, don’t place important buttons where they’ll be blocked by a pop-up window. The user can’t press a covered button. For details on where windows appear, refer to Chapter 11, *Configuring Alarms,* and Chapter 12, *Configuring Terminal Setup Options.*

The PanelView 1000e and 1400e touch screen terminals use analog-resistive touch screens. This allows input objects to be any size (minimum 40 by 40 pixels) and to be placed anywhere on the screen. The analog-resistive touch screen provides greater flexibility in screen design compared to PanelView 1200 and 1200e terminals. PanelView 1200 and 1200e terminals use a touch matrix, which requires that all input objects align with a touch grid.

ATTENTION: Do not press multiple touch screen locations at the same time. In the unlikely event that two or more touches occur simultaneously in the same horizontal or vertical line, an input object located half-way between the two touches could be activated. This is inherent in the analog-resistive touch screen technology used in PanelView 1000e and 1400e terminals.

- Ensure there is always a clear way to move between screens, especially if the programmable controller doesn't control the screen changes.

Use a Screen Template to Ensure Consistency

To help increase operator efficiency, keep a consistent appearance between all the screens in an application. Present the same pieces of information in the same place on each screen, for example:

- titles
- "Goto Screen" buttons
- local messages and ASCII display objects
- time and date objects

A screen template is a useful tool for ensuring consistency between screens.

To create a screen template:

1. Use a screen design worksheet to hand-draw a screen template that contains the elements common to all or most screens.

   Photocopy the 1000e, 1200e, or 1400e screen worksheet in Appendix A for this purpose. Save the original for later use.
2. Make a photocopy of the hand-drawn template for each application screen.

3. Design each application screen on a copy of the screen template.

You can also design the screen template in PanelBuilder, copy it each time you need to create a new screen, and use it as your starting point for each new screen. For more information, see Chapter 8, *Creating Screens*.

Using Worksheets to Design Screens

PanelBuilder and the Windows environment offer considerable flexibility for creating and changing application screens. Nevertheless, you may find it useful to plan the screens using screen worksheets before you begin developing the application.

Use the following worksheets to define screens and objects:

- Screen Worksheet for 1000e Terminals
- Screen Worksheet for 1200e Terminals
- Screen Worksheet for 1400e Terminals

Choose the worksheet that matches the PanelView terminal on which the application will run. Both show the location of the terminal function keys relative to the screen. Each worksheet can be used for designing either touch screen or keypad applications.

Screen Worksheet for PanelView 1200e Terminals

Use this worksheet to plan application screens for PanelView 1200e keypad and touch screen applications.

The position of the touch cells is marked on the worksheet for you to use as a guide in positioning touch cell input objects. The screen area is 640 by 480 pixels in size. There is a 10 by 12 grid of touch cells; the smallest size for a touch object is 64 by 40 pixels. PanelBuilder sizes touch objects only in multiples of touch cells.

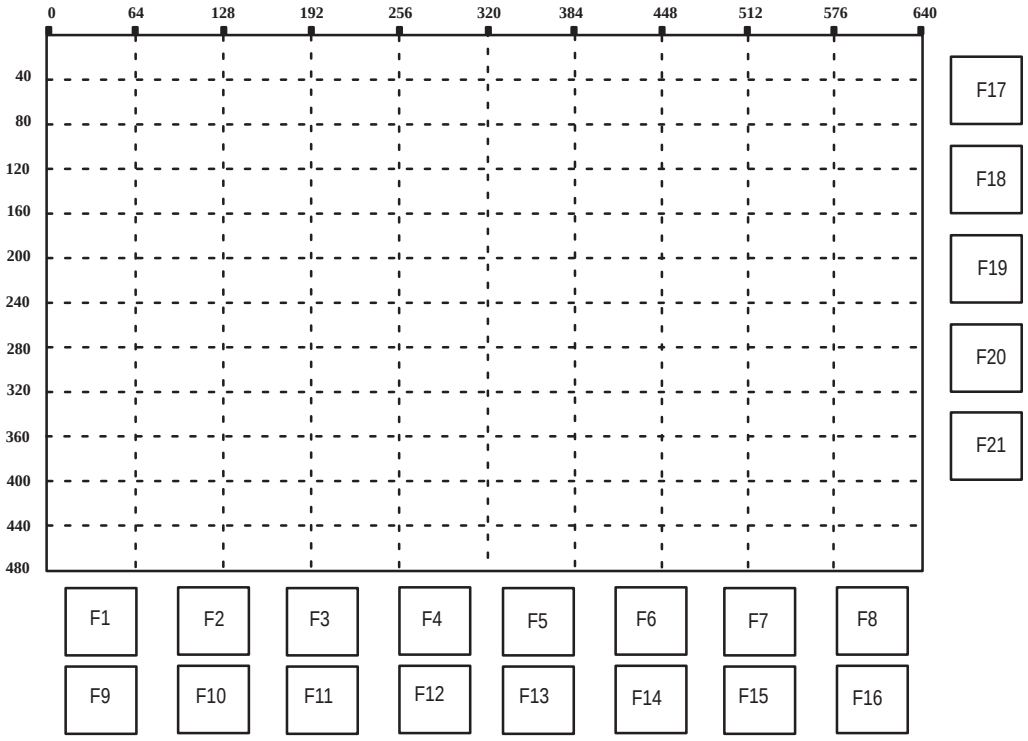
The position of the terminal function keys is marked on the worksheet for you to use as a guide in positioning function key input objects. Objects can be placed anywhere on the screen; they need not align to this grid.

Important: Non-input objects can overlap, but cannot be positioned off-screen.

Screen Worksheet for 1200e Terminals

Application File Name: _____

Screen Number: _____ **Screen Name:** _____



Notes:
1. The grid shows the location of the touch cells on a PanelView 1200e Touch Screen terminal. PanelBuilder sizes buttons in multiples of touch cells. The smallest size for a 1200e touch screen application input object is 64 x 40 pixels.
2. You can use this worksheet to align buttons you draw on the screen to keypad terminal function keys.

To complete each PanelView 1200e screen worksheet:

1. Draw all objects that will be displayed on each screen. Most objects can be any size.

Note: The smallest size for a PanelView 1200e touch object is 64 by 40 pixels. PanelBuilder sizes 1200e touch objects only in multiples of touch cells.

2. Make a note of the button labels on the worksheet. This will help you remember where you are locating common objects such as GoTo Screen buttons and Exit buttons.

Screen Worksheets for PanelView 1000e or 1400e Terminals

Use these worksheets to plan application screens for PanelView 1000e or 1400e keypad and touch screen applications.

The position of the terminal function keys is marked on the worksheet for you to use as a guide in positioning your application screen function key input objects.

The screen area is 640 by 480 pixels.

The PanelView 1000e and 1400e touch screen terminals use analog-resistive touch screens. This allows input objects to be any size (minimum 40 by 40 pixels) and to be placed anywhere on the screen. The analog-resistive touch screen provides greater flexibility in screen design compared to PanelView 1200 and 1200e terminals. PanelView 1200 and 1200e terminals use a touch matrix, which requires that all input objects align with a touch grid.

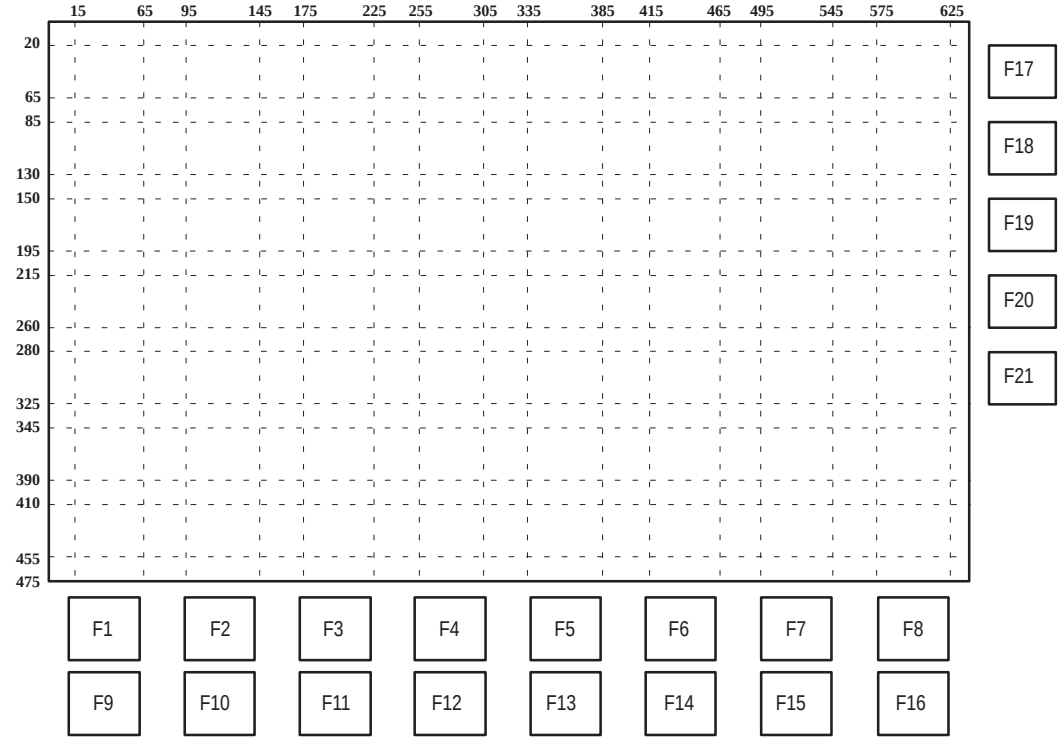
ATTENTION: Do not press multiple touch screen locations at the same time. In the unlikely event that two or more touches occur simultaneously in the same horizontal or vertical line, an input object located half-way between the two touches could be activated. This is inherent in the analog-resistive touch screen technology used in PanelView 1000e and 1400e terminals.

Important: Non-input objects can overlap, but cannot be positioned off-screen.

Screen Worksheet for 1400e Terminals

Application File Name: _____

Screen Number: _____ **Screen Name:** _____



Notes:
1. You can use this worksheet to align buttons you draw on the screen to keypad terminal function keys.
2. The PanelView 1400e touch screen terminals use analog-resistive touch screens. This allows input objects to be any size (minimum 40 by 40 pixels) and to be placed anywhere on the screen. The analog-resistive touch screen provides greater flexibility in screen design compared to PanelView 1200 and 1200e terminals. PanelView 1200 and 1200e terminals use a touch matrix which requires that all input objects align with a touch grid.

To complete each PanelView 1000e or 1400e screen worksheet:

1. Draw all objects that will be displayed on each screen. Most objects can be any size.

Note: The smallest size for a touch object is 40 by 40 pixels.

2. Make a note of the button labels on the worksheet. This will help you remember where you are locating common objects such as GoTo Screen buttons and Exit buttons.

Step 3: Plan Communications with the PLC

The third step in designing an application is to determine the programmable controller addresses relevant to your application. Because the tasks involved in planning PLC addresses differ for Remote I/O and DH+ or ControlNet applications, the remainder of this chapter is divided into two sections, one for each.

To continue planning a DH+ or ControlNet application, go to Page 2–35.

Planning Remote I/0 Communications

This section describes how to complete Step 3 for a Remote I/O application. It includes information about these tasks:

- Identifying inputs and outputs
- Planning rack assignments
- Planning addresses
- Using worksheets to help you plan Remote I/O communications

Identifying Inputs and Outputs

The first step in determining which addresses the application will need is to identify the inputs and outputs the process requires. Based on that information, you can determine the number and size of the PLC racks required.

To identify inputs and outputs, you need to know how addressing works on the programmable controller. This section provides information to help you understand:

- addressing
- rack assignments
- module groups
- shared racks

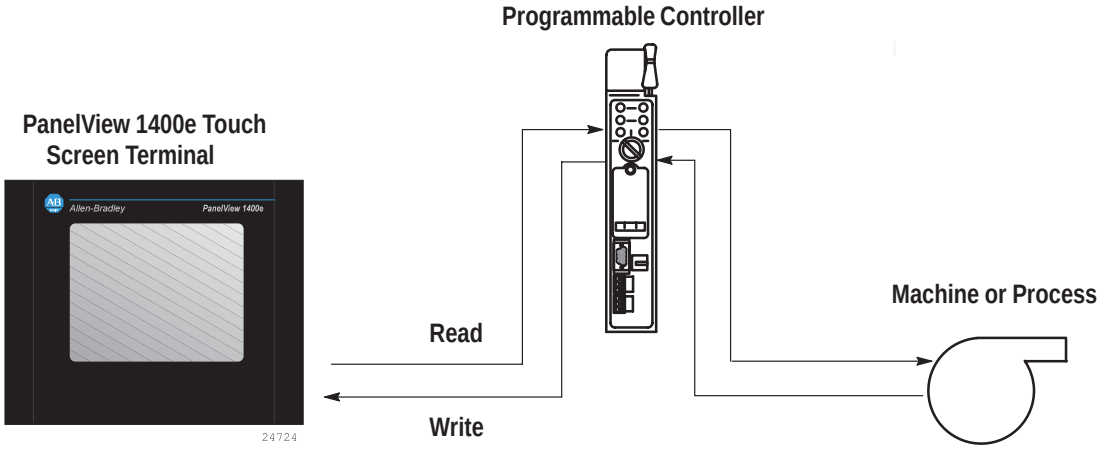
Understanding Addressing

When you assign tags to objects, windows, and PLC I/O Control options, you are assigning an address. You need to assign the tag to establish communication between the programmable controller and the application in the PanelView terminal.

When the application is running, it sends information to and receives information from the programmable controller. The controller communicates through the input and output addresses assigned to the tags.

The addresses defined in the tags work as follows:

- the input addresses are for inputs to the programmable controller (for example, control list selectors, push buttons, or other objects that write to the programmable controller)
- the output addresses are for outputs from the programmable controller (for example, display, current state, alarms, or counts from the programmable controller)



Addresses, defined as racks, words and bits, are locations within the Remote I/O racks. These locations are represented on the PLC's Remote I/O image table.

The two types of addresses objects can use are:

- discrete Remote I/O addresses
- block transfer file addresses

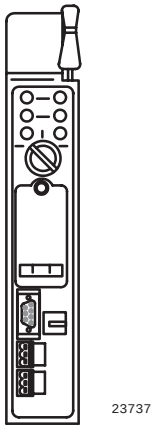
These two address types are discussed in detail later in this chapter.

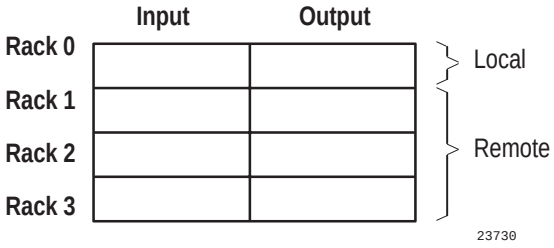
Understanding Rack Assignments

PanelView terminals are connected to a host controller, usually a programmable controller.

The programmable controller communicates with one or more I/O racks to solicit its input and write its output. An I/O rack is an addressing unit that corresponds to up to eight input words and eight output words in the I/O image table. To the programmable controller, the PanelView terminal looks like any other Remote I/O rack. You can configure the PanelView terminal as one or more full or partial Remote I/O racks in the Rack editor in PanelBuilder. Depending on the type of controller, the PanelView terminal can simulate up to 64 different rack numbers. It can also simulate partial racks.

The following figure shows a PLC 5/15. It is configured for one local and three Remote I/O racks. The PanelView terminal could be configured to occupy all or part of the three Remote I/O racks.





Programmable Controller (PLC 5/15)

For detailed information on configuring I/O racks, refer to Allen-Bradley Programmable Controller and Remote I/O Scanner user manuals.

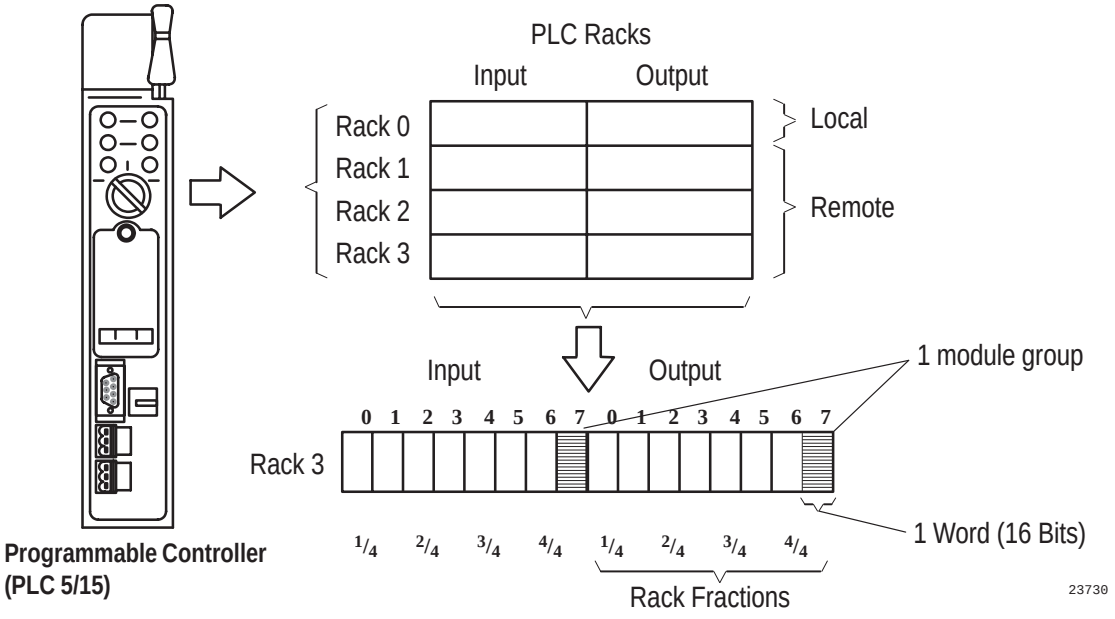
Understanding Module Groups

A PanelView rack supports up to eight module groups, numbered 0–7. A module group consists of a 16-bit input word and a 16-bit output word. The following table shows the starting module groups for the four rack sizes:

With this starting module group		You can assign these rack sizes			
1st quarter of rack	0	1/4	1/2	3/4	FULL
2nd quarter of rack	2	1/4	1/2	3/4	
3rd quarter of rack	4	1/4	1/2		
4th quarter of rack	6	1/4			

Because the smallest part of a rack you can assign is 1/4, you must assign module groups in pairs: 0,1; 2,3; 4,5; and 6,7. Each pair supports 32 input bits and 32 output bits.

The following figure expands the previous illustration with the module groups for rack 3. It shows that module group 7 is made up of input word 7 and output word 7. The module group pair 6,7 uses the fourth quarter of the rack.



Important: Each separate or partial rack assignment adds 5 to 7 milliseconds to the Remote I/O scan time. To maximize speed, assign the PanelView terminal to as few PLC racks as possible.

Understanding Shared Racks

If the PanelView terminal doesn't occupy a full rack on the programmable controller, another PanelView terminal can share the same rack by occupying the unused partial Remote I/O rack. Each PanelView terminal with full access must occupy a unique section of the rack.

The following table is an example of how two Remote I/O racks can occupy the same PLC rack:

Parameter	First Remote I/O Rack	Second Remote I/O Rack
Rack Number	1	1
Starting Module Group	0	6
Rack Size	3/4	1/4
Last Chassis	No	Yes

PanelView uses the Last Chassis designation to optimize PLC scanning. When you specify a partial rack definition as the last chassis, the programmable controller won't scan beyond the specified module group on the rack.

Understanding Full and Listen Only Access

Full access means that the PanelView terminal has full read and write access to the rack's I/O space. Listen Only access means that the PanelView terminal can read from the rack's I/O space, but cannot write to it. Listen Only is useful when one PanelView application wants to monitor the I/O activity on another I/O rack without having the programmable controller handle or re-route the information.

Planning Rack Assignments

To plan your application fully, you need to determine the racks and partial racks you will use. The method for planning your racks depends on these factors:

- type of programmable controller
- size of application you will create
- rack space available

When you create an application, you assign tags to dynamic objects, windows, and PLC I/O Control options. These tags define the programmable controller addresses to use. You can assign two kinds of addresses for a tag:

- discrete I/O addresses
- block transfer file addresses

To maximize the efficiency of your application, plan the addresses before you begin creating the application.

Note: You can also assign read tags to expressions. Read tags display output from the PLC. You can use expressions to manipulate the value of the read tag and display the result on the PanelView terminal. See Chapter 7, *Creating Expressions,* for more information.

Discrete I/O Addresses

Discrete I/O addresses communicate directly with the PLC. They provide faster updates than block transfer file addresses, but require more rack space. Therefore, keep the following objects in discrete I/O:

- all push buttons
- all states or values that you want updated immediately

Because there is only a limited number of I/O points, you might not be able to use discrete I/O addresses for your entire application. If you do not have enough rack space, create block transfer files.

Block Transfer File Addresses

A block transfer file transfers information to and from the PLC through a control byte in the discrete output space. If you don't have much rack space, you can create up to 64 block transfer files. Block transfer files give you more input and output space, but are not as fast as using discrete I/O.

Each PanelView terminal can have up to 64 block transfer files up to 64 words in length. Block transfer files can be defined as either read or write files. You can choose the racks and modules that the block transfer control bytes will occupy and specify the size of the file.

Each block transfer file requires one byte (8 bits or 1/2 word) of discrete output on the PLC to control it. You must choose which byte in the PLC word will support each block transfer. Once you've assigned a block transfer file to a byte, you cannot assign a discrete address or the Pass-Through file to that output byte. A single control byte can control both a block transfer read and a block transfer write.

Use block transfer file addresses for objects, windows, and PLC I/O Control options.

Note: If you are using floating point data in your application, you need to define one or more block transfer files.

The PanelView terminal references these addresses according to:

- block transfer file number (B1 to B64)
- word number (0 to 63)
- bit number (0 to 15)

Important: The SLC-5/02 and some other PLCs do not support block transfer file addressing. For more information, see the applicable programmable controller manuals.

Determining Rack Size and Block Transfer File Requirements

Rack size and block transfer requirements depend on the number of objects, windows, and PLC I/O Control options that must interact with the PLC.

The smallest rack size you can assign is two module groups, or 1/4 rack. If this size is not enough to support your application features, do any or all of the following:

- increase the rack size to 1/2, 3/4, or FULL
- assign additional rack numbers
- assign block transfer files

To determine the rack sizes you will need full access to, find out:

- what rack numbers are available. This depends on the type of PLC you are using.
- approximately how many racks you will need for your application

You must also decide which racks require monitoring only. Define these with Listen Only access.

Use the following table as a guide to determine if you'll need to use block transfer files. The table shows, per rack, the number of discrete I/O image table words and bits available for PLC communication.

Note: You can assign up to 64 rack numbers.

Rack Size	16-bit Input Words	16-bit Output Words	Total Input Bits	Total Output Bits
1/4	2	2	32	32
1/2	4	4	64	64
3/4	6	6	96	96
FULL	8	8	128	128

Each block transfer file provides up to 64 words (1,024 bits) of space. The file can be either read or write. You can define up to 64 block transfer files.

Note: Each block transfer file occupies one byte of discrete output, which reduces the amount of available discrete I/O.

Reducing PLC Addressing Requirements

You can reduce PLC address requirements by multiplexing. Multiplexing means using one screen object for more than one purpose depending on the state of a variable, or using one address for multiple objects on different screens.

Normally you won't need to multiplex objects because the PanelView terminal provides plenty of separate I/O addresses, plus the capability for block transfers and multiple racks. But if your application requires extra memory, you can multiplex momentary push buttons, numeric display data, and other objects.

Important: Multiplexing increases the complexity of PLC programs, which can outweigh the benefits of address space saved. Use multiplexing with caution.

Planning Addresses

After designing the screens for your application, you know the types and numbers of objects you will need. At this point you can plan the object addresses.

The addresses used throughout the application are defined in the tag database. Each entry in the database is called a tag, which is simply a symbolic name for an address. The tag database provides a way to define all addresses in one place. The same address need only be defined once no matter how many objects use the address. To assign an address to an object, assign a tag, or assign an expression that manipulates the value of one or more read tags before displaying the result on the PanelView terminal. For information about expressions, see Chapter 7, *Creating Expressions*.

Using Worksheets to Plan Addresses

Use the following worksheets to keep track of the objects and the amount of address space being used. Appendix A contains sample worksheets. Fill them in at the same time.

- **Remote I/O Tags Worksheet**—Use the Remote I/O Tags worksheet to record information about all dynamic objects. Dynamic objects are those that communicate with the PLC. For a list of dynamic objects, see Chapter 9, *Creating Objects*.
- **Discrete I/O Usage Worksheet**—Use this worksheet for discrete file addresses. Fill it in as you fill in the Remote I/O Tags worksheet, to keep track of free and allocated PLC I/O addresses.
- **Block Transfer File I/O Usage Worksheet**—Use this worksheet for block transfer file addresses. Fill it in as you fill in the Remote I/O Tags worksheet, to keep track of free and allocated PLC I/O addresses.

Once you have filled in the previous three worksheets, use the next two worksheets to complete your address planning.

- **Rack Assignments Worksheet**—Use this worksheet to keep track of your rack assignments.
- **Block Transfer File Worksheet**—Use this worksheet to define block transfer file addresses and the Pass-Through control byte address.

These points summarize the process of planning your Remote I/O communications using the worksheets:

- Define the discrete or block transfer PLC addresses and tag names for each object on your screen using the Remote I/O Tags Worksheet.
- Enter the object reference number and screen number on the I/O Usage worksheets to map your PLC memory usage. Use the Discrete I/O Usage Worksheet for discrete addresses. Use both the Discrete I/O Usage Worksheet and the Block Transfer File I/O Usage Worksheet for block transfer files and addresses; the Discrete I/O Worksheet is for recording any block transfer control bytes in the Output section of the image table.
- Use a new Remote I/O Tags Worksheet for each screen. Use one Discrete I/O Usage Worksheet for each full or partial rack assignment, and two Block Transfer File I/O Usage Worksheets for each block transfer file.
- Record the addresses for your windows and PLC-controlled options on a different worksheet, and map these addresses onto the Discrete I/O Usage Worksheets.
- Finish assigning addresses to your objects, windows, and PLC-controlled options so you know how many racks to assign to your PanelView terminal. Enter this information on the Rack Assignments and Block Transfer File Worksheets.

  Important: Before you use the worksheets, determine the rack numbers you can use for your PanelView terminal. You won't know your exact PLC I/O address requirements until you've defined all the objects, windows, and PLC I/O Controlled Options. However, you do need to know the rack numbers available so you can plan the PLC addresses.

The Remote I/O Tags Worksheet

For each application screen, use one or more Remote I/O Tags Worksheets to record tag information for each dynamic object. The Remote I/O Tags Worksheet is available in Appendix A. Only a few objects require that you fill in all the columns.

Also use Remote I/O Tags Worksheets to record information about Alarm and Information Message windows, and PLC I/O Control options. These items are not specific to a particular screen, so use a separate worksheet for them.

Remote I/O Tags Worksheet

Application File Name: _____ **Screen Number:** _____

Object			Tag							Address		
Ref No.	Type	Initial State Value	Name	Type	Analog Tag only: Scale	Analog Tag only: Offset	Analog Tag only: Data Type	Initial Value	Range of Values	Start Address	No. of Bits, End Address or (string tag only) No. of characters	Tag Start Value

Address formats:

Discrete: I/O:rrw/bb I is input, O is output; rr is rack number; w is start word; /bb is optional bit offset. All numbers are octal. e.g., I:011/12
Block Transfer: Bff ww/bb B is block transfer; ff (followed by a space) is file number; ww is start word; /bb is optional bit offset. All numbers are octal. e.g., B1 01/01
Discrete SLC: I/O:ss.ww/bb I is input, O is output; ss is optional slot number; ww is start word; /bb is optional bit offset. All numbers are decimal. e.g., O:01.18

Notes:

Initial State Value: Objects such as Multistate Push Buttons have a user-configurable initial state whose value will supersede the tag's initial value when the application is first run. This is called the tag's start value.
Tag Start Value: The start value is the tag initial value, or the object initial state value, if one exists.

Object		
Ref No.	Type	Initial State Value

Recording Object Information

For each object, fill in the following information:

- **Ref No.:** The object's unique identification number. Usually, you can start with 1 and increase by 1 for each object. Start from 1 again for each screen so that adding an object to the first screen doesn't require that all other numbers change.

- **Type:** The type of object, such as Interlocked Push Button.

  Some objects use several tags or expressions. See online Help or the *PanelBuilder 1400e Configuration Software for Windows Reference Manual*, or look at the PanelBuilder objects themselves, to plan which tags or expressions to use for each object.

  Use one line in the Remote I/O Tags worksheet for each tag and use the Type column to describe the function of the additional tags.

  Expressions are optionally used to read and manipulate the PLC output at the tag addresses. You can also use a line for each expression, to plan which ones you will need to define when you create the object. For information about expressions, see Chapter 7, *Creating Expressions*.
- **Initial State Value:** Objects such as Multistate Push Buttons have a user-configurable initial state whose value will supersede the tag's initial value when the application is first run.

Tag						
Name	Type	Analog Tag only			Initial Value	Range of Values
		Scale	Offset	Data Type		

Recording Tag and Expression Information

The tag database records all the addresses used throughout an application. Instead of assigning an address directly to an object, you enter the address as a tag definition in the database, and then assign the tag, or an expression that uses one or more tags, to an object. If the same address is used by more than one object, you need only define it once in the database. For information about expressions, see Chapter 7, *Creating Expressions*.

Some objects require more than one control tag or expression. For example, a Maintained Push Button needs a Button Control tag and an Indicator State tag or expression. Record all necessary tag names and addresses in the worksheet, as well as information about the expressions.

For each expression, fill in the name of the expression in the object (for example, Indicator State), the name of the tag or tags the expression will read data from, and the manipulation the expression will perform on the tag data. For information about the kinds of operations expressions can perform, see Chapter 7, *Creating Expressions*.

For each tag, fill in the following:

- **Name:** A symbolic name you assign to the address. The tag name can be up to 40 characters long and can include letters, numbers, the dash (-), and the underscore character (_). Spaces and periods (.) are not permitted.

When you define tags in the Tag Database editor, you can also define folders. Folders are a convenient way to organize tags. For example, you can put all the tags for one screen's objects in one folder. The folder name becomes part of the tag name, and is separated from the rest of the name with a backslash. For example, the full name for the tag "StartButton_Control" in the folder called "Screen_1" is "Screen_1\StartButton_Control".

For details on creating tags and folders, see Chapter 6, *Defining Tags*.

- **Type:** Specify one of these types of tags:

Tag Type	Type of Data Stored
Digital	A single bit, giving the values 0 or 1. Use a digital tag for devices that can only be on or off, such as push buttons.
Analog	Range of values (depends on the data type selected). Use an analog tag to control or monitor a value, for example, a temperature or the position of a rotary control. When you define an analog tag, you must also fill in the Scale, Offset and Data Type fields.
String	ASCII string, series of characters, or whole words (maximum of 82 characters). Use a string tag only with the ASCII Read and ASCII Display objects, and (optionally) for sending alarm messages to the PLC.
Block	A contiguous bit string of up to 1024 bits. A block tag is used strictly for defining the Alarm Triggers and Acknowledge to PLC controls. These controls can monitor and acknowledge multiple simultaneously active alarm conditions. For details, see Chapter 11, *Configuring Alarms*.

- **Scale (Analog Tag Only):** Specify a value, with 1 being the default. The scale and offset work together to modify any raw data that comes from or is written to a PLC. (For example, to convert the raw data into engineering units.)
- **Offset (Analog Tag Only):** Specify a value, with 0 being the default. The scale and offset work together to modify any raw data that comes from or is written to a PLC.

  When the PanelView terminal reads a value from the PLC, it multiplies this value by the scale value and then adds the offset before it writes the value to the terminal's value table. Similarly, when the terminal sends data from the terminal to the PLC, it first subtracts the offset and then divides by the scale value to convert the value back to raw data before it is written to the PLC.

  With Scale = 1 and Offset = 0 (the default values), PLC data is not changed.

Note: For PLC output from Analog and Digital tags, you can create expressions to manipulate or convert the data as needed.

- **Data Type (Analog Tag Only):** The type of data that the PLC address will accept.

  Note: Not all PLCs accept all data types. For the list of valid data types for the PLC you are using, see your PLC's User Manual.

  Specify one of the following:

  - **Default:** The default data type is an unsigned integer.
  - **Unsigned Integer:** An unsigned integer occupies one 16-bit word, stored in binary format. The range of raw data values is from 0 to +65535.
  - **Integer:** A signed integer occupies one 16-bit word, stored in two's complement format. The range of raw data values is from –32,768 to +32,767.
  - **Long Integer:** A long integer occupies two 16-bit words, stored in two's complement format. The range of raw data values is from –2,147,483,648 to +2,147,483,647.
  - **Floating Point:** Single-precision (32-bit) floating point. The range of data values is –3.402823E +38 to –1.175494E –37, 0, and +1.175494E –37 to +3.402823E +38. For Remote I/O applications, the tag must use a Block Transfer File address.
  - **Byte:** The byte data type occupies 8 bits of unsigned data. The range of raw data values is from 0 to 255.
  - **Binary:** When this data type is assigned, the object's PLC address will be a contiguous bit string 1 to 16 bits long. The number of bits you assign determines the range of values you can access.

    The range of values is 0 to $(2^n) – 1$ where n is the number of bits. For example, the range of values for 16 bits is 0 to 65535.

    You can position a binary data address anywhere within a PLC word. The binary data address can extend into the next word, but cannot extend past the end of a rack assignment or block transfer file.

    The remaining bits in that word, on either side of the binary data address, can be used for other tags with other data types.

    The following table shows the relationship between binary states and bits:

This number of binary states	Requires this many bits
1–2	1
3–4	2
5–8	3
9–16	4
17–32	5
33–64	6
65–128	7
129–256	8

This number of binary states	Requires this many bits
257–512	9
513–1024	10
1025–2048	11
2049–4096	12
4097–8192	13
8193–16384	14
16385–32768	15
32769–65536	16

- **Bit Position:** This data type is a contiguous bit string. The position of the least-significant on bit in the bit string determines the value of the tag.

  The number of bits required depends on the maximum number of states. Use this formula to determine the number of bits an object requires:

  Number of bits = (Maximum State Value – Minimum State Value)

  This data type can also be used for objects controlled by PLC outputs. For example, you could configure a Multistate Indicator to have 12 states with state values 0 to 11, each with unique text and attributes. The PLC output address would then be an 11-bit string. The bit set by the PLC would determine the state displayed by the terminal.

  Important: If more than one bit is set at a time, the terminal considers only the lowest bit set.

- **BCD (Binary Coded Decimal):** If you assign one of the BCD data types, the object's PLC control word will be a contiguous bit string of one of these numbers of bits:

This number of BCD digits	Requires this many bits
1 BCD digit	4
2 BCD digits	8
3 BCD digits	12
4 BCD digits	16
5 BCD digits	20
6 BCD digits	24
7 BCD digits	28
8 BCD digits	32

Each group of four bits represents the equivalent of one decimal digit (0 to 9) in the tag value. The number of bits required depends on the maximum value required for the object. This depends on the application.

You can align the BCD bit string on any PLC word bit. Most of the time you'll want to start at bit 0. The BCD bit string can extend into the next word, but cannot extend past the end of a rack assignment or block transfer file.

Example: BCD Values or States

1. If you configured a BCD data value to have a maximum of 16 states (2 digits), the assigned address would be an 8-bit contiguous string with 4 bits for each digit.

   2 digits @ 4 bits each = 8 bits

2. If a numeric value display was configured to have a range from 00000 to 99999 (5 digits), the assigned address would be a 20-bit contiguous string with 4 bits for each of the 5 digits.

   5 digits @ 4 bits each = 20 bits

   A PLC word can support only 16 bits, so an additional word is required for the additional 4 bits.

3. When more than one PLC word is needed to support the required number of digits, all remaining digits extend into the next word. For example, BCD value 123456 would be represented in two PLC words as follows.

Bit Numbers (octal)	17 16 15 14	13 12 11 10	7 6 5 4	3 2 1 0
Word X	3	4	5	6
Word X + 1			1	2

Note: The BIN3, BIN4, BIN6, and BIN8 data types are reserved for use in Modbus applications. For more information, see the *PanelBuilder 1400e Modbus User Manual*, Publication Number 2711E-6.12, which is supplied as part of the optional Modbus Communications Kit, Catalog Number 2711E-UMOD.

- **Initial Value:** The value to be placed in the input address of a tag on power-up. As with any value sent from the terminal to the PLC, this value will have the offset subtracted and then be divided by the scale. Assign Initial Values to tags with PLC input addresses.

  Note: For any multistate object you draw in the Screens editor, you can define an Initial State. If you define an Initial State for the object and an Initial Value for the object's control tag, the object's Initial State takes precedence. The tag's initial value is used only if an initial state is not defined.

  Note: A tag can have only a single initial value. If more than one object is configured to write its initial state value to a single tag, a validation error will occur if the initial state values are different.

- **Range of Values:** If an object has a range of state values, record the values in this column.

Range of Values

Recording the Address

For each tag, record the PLC address as follows:

Start Address

- **Start Address:** For Remote I/O applications, the starting point of the address. To address an area of memory larger than a single bit, specify the length or end address.

  The syntax depends on the type of address or controller. See "Remote I/O" in Appendix C, *Addressing Syntax*.

No. of Bits, End Address or (string tag only) No. of characters

- **No. of Bits, End Address or (String Tags Only) No. of Characters:** The length of the tag's address. Specify one of the following:
  - For digital tags, the bit address you enter in the Start Address column is the entire address. Ignore this column.
  - For analog tags, specify either the number of bits used or the end address. In the Tag Database editor, you can enter this address in one of two forms:

    Start Address, No. of Bits (that is, the start address, a comma, and the number of bits, like this: I:011/01,3)

    Start Address–End Address (that is, the start address, a dash, and the end word and bit, like this: I:011/01–11/04)
  - For String tags, specify the length of the string in characters.
  - For Block tags, specify the number of bits, up to 1024. If the tag uses a discrete I/O start address, the number of bits allowed is 1 to 128. Use a block transfer file for longer block lengths. See Chapter 11, *Configuring Alarms*, for information about how block tags work to trigger and acknowledge alarms.

Tag Start Value

- **Tag Start Value:** The start value is the tag initial value, or the object initial state value, if one exists.

The Discrete I/O Usage Worksheet

Complete the Discrete I/O Usage Worksheet at the same time as you complete the Remote I/O Tags Worksheet, so you can keep track of free and allocated PLC I/O addresses.

The Discrete I/O Usage Worksheet shows the PLC I/O image table. Use one Discrete I/O Usage Worksheet for each full or partial rack assignment, marking the PLC control bit(s) used in the tags assigned to each object, window, or PLC I/O Control option.

Discrete I/O Usage Worksheet

Application File Name: _____ Rack Number:_____

Full or Listen Only Access (Circle One)

PLC Output Bit Numbers

	17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0
0																
1																
2																
3																
4																
5																
6																
7																

PLC Input Bit Numbers

	17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0
0																
1																
2																
3																
4																
5																
6																
7																

The Block Transfer File I/O Usage Worksheet

Complete the Block Transfer I/O Usage Worksheet at the same time as you complete the Remote I/O Tags Worksheet, so you can keep track of free and allocated PLC I/O addresses.

Complete both sheets for each block transfer file you use, marking the PLC control bit(s) used in the tags assigned to each object, window, or PLC I/O Control option.

Block Transfer File I/O Usage Worksheet (Page 1 of 2)

Application File Name: _____

Block Transfer File Number: _____ (Enter a number between 1 and 64)

Number of Words: _____ (Enter a number between 1 and 64)

READ (PLC In) or **WRITE** (PLC Out) (Circle One)

FULL or **LISTEN ONLY** Access (Circle One)

PLC Bit Numbers

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0																
1																
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																
17																
18																
19																
20																
21																
22																
23																
24																
25																
26																
27																
28																
29																
30																
31																

Block Transfer File I/O Usage Worksheet (Page 2 of 2)

Application File Name: _____

Block Transfer File Number: _____ (Enter a number between 1 and 64)

Number of Words: _____ (Enter a number between 1 and 64)

READ (PLC In) or **WRITE** (PLC Out) (Circle One)

FULL or **LISTEN ONLY** Access (Circle One)

PLC Bit Numbers

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
32																
33																
34																
35																
36																
37																
38																
39																
40																
41																
42																
43																
44																
45																
46																
47																
48																
49																
50																
51																
52																
53																
54																
55																
56																
57																
58																
59																
60																
61																
62																
63																

When you have assigned addresses to all the tags used for each object, window, and PLC I/O Control option, determine the I/O requirements. The best way to do this is to record the information on the Rack Assignments and Block Transfer File worksheets.

The Rack Assignments Worksheet

Once you know how much I/O your application requires, use the Rack Assignments Worksheet to keep track of your rack assignments.

Rack Assignments Worksheet

Application File Name: _____

PLC/Scanner Type: _____

Baud Rate: 57.6K (10,000 Ft) 115.2K (5,000 Ft) 230.4K (2,500 Ft)

Note: You must make at least one rack assignment. Additional rack assignments are optional. Record the Rack Number and circle the Module Groups, Last Chassis designation, and Access used. Use additional worksheets if necessary.

Rack Assignments

Rack Number	Module Groups	Last Chassis	Access
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only
	0,1 2,3 4,5 6,7	Y N	Full Listen Only

Recording the PLC/Scanner Type

In the PLC/Scanner Type field, record the programmable controller or scanner type that you will be using. The following table lists the available programmable controllers and scanners:

Programmable Controller	Scanner
PLC-2/20 or PLC-2/30	1772-SD/SD2
PLC-3	1775-S4A/B
PLC-3	1775-S5
PLC-3/10	1775-SR5
PLC-5/11	Integral
PLC-5/15	Integral
PLC-5/20	Integral
PLC-5/25	Integral
PLC-5/30	Integral
PLC-5/40	Integral
PLC-5/60	Integral
PLC-5/80	Integral

Programmable Controller	Scanner
PLC-5/250	5150-RS
SLC 5/02, 5/03 or 5/04	1747-SN Series A or Series B
IBM PC	6008-SI
VME Bus	6008-SV
DEC Q-Bus	6008-SQ
Other	1771-SN Sub Scanner

Selecting the Baud Rate

The baud rate determines the speed at which the terminal communicates with the programmable controller. The baud rate depends on the cable length, and on the baud rate supported by other devices on the Remote I/O network.

Select this baud rate	For distances up to
230.4K	2,500 feet
115.2K	5,000 feet
57.6K	10,000 feet

Note: The PLC-5/15 and PLC-5/25 support only 57.6K baud.

Assigning Racks

For each assignment in the Remote I/O Tags Worksheet, define the following:

- **Rack Number:** Specify the number of the rack you're using.
- **Module Group:** Circle the module groups you are assigning to the rack number.
- **Last Chassis:** Choose Yes if there are no other fractional PLC rack assignments that have the same rack number and a higher starting module group number. Otherwise choose No.
- **Access:** Choose Full if the PanelView terminal will have full read and write access to the rack; choose Listen Only if the terminal will only monitor the rack.

The Block Transfer File Worksheet

Block transfer files occupy space in the PLC discrete output table only. For each file, define the block transfer file address and the block transfer file control address in the PLC.

Important: Some scanners (the 1771-SN and 1747-SN Series A Modules, for example) do not support block transfers to a Remote I/O device. Block transfer file assignments apply only if you're using another Remote I/O scanner device. See your scanner's User Manual to determine whether it supports the use of block transfers.

Block Transfer File Worksheet

Application File Name: _____

PLC/Scanner Type: _____

Baud Rate: 57.6K (10,000 Ft) 115.2K (5,000 Ft) 230.4K (2,500 Ft)

Notes:

1. Block Transfers are optional. Record the Block Transfer File number and size, write in one of the rack numbers from the previous worksheet, circle the appropriate Word and Byte, circle Read or Write, and Access used. Use additional worksheets if necessary.
2. If the Block Transfer File is on a Listen Only rack, the file can only have Listen Only access.
3. The Pass-Through Block Transfer File must be defined on a Full Access rack.
4. For each block transfer file, mark off the byte in the output image on the Discrete I/O worksheet.

Block Transfer File Assignments

Block Transfer File Number	Size	Rack	Word	Byte	Usage	Access
Pass-Through	64		0 1 2 3 4 5 6 7	Hi Lo	Both	Full
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only

Using the Remote I/O Tags and Rack Assignments worksheets as a reference, define the following:

- **Block Transfer File Number:** Specify the file number (B1 to B64).
- **Size:** Specify the number of words required (1 to 64).
- **Rack:** Specify a rack number.
- **Word:** Assign the word that will contain the block transfer control byte.
- **Byte:** Choose Hi for high control byte or Lo for low control byte.
- **Usage:** Specify whether the control address is for a block transfer read or block transfer write.
- **Access:** Specify full or listen-only access as determined by the rack definition.

Assigning Pass-Through

The Pass-Through option provides a way to transfer applications remotely, to and from the PanelView terminal, using the DH+ and Remote I/O, ControlNet and Remote I/O, or Ethernet and Remote I/O networks. You can assign the Pass-Through address using PanelBuilder, enabling you to transfer an application file without taking the PanelView terminal out of Run mode.

The Pass-Through block transfer file uses a control byte in the output image table, which cannot be shared with any other block transfer file or discrete tag. The control byte must be defined on a rack that has full access.

Mark the Rack number, Word, and Byte on the Block Transfer File Worksheet. If you are using an SLC 5/04 device, also note the SLC Slot number in the Rack column.

Carrying Out Your Plan

After you complete the planning steps in this chapter, your worksheets contain all the information you need to create your application. To create your application, work through the remaining chapters in this manual, as described in “Steps for Building an Application,” at the beginning of this chapter.

Planning DH+ or ControlNet Communications

This section describes how to complete Step 3, planning how the application will communicate with the PLC, for a DH+ (Data Highway Plus) or ControlNet application. It includes information about these tasks:

- using worksheets to help you plan DH+/ControlNet applications
- defining nodes
- defining scan classes
- planning addresses

Using Worksheets to Plan Communications

DH+ and ControlNet

Use these worksheets to determine addressing and communication requirements in DH+ and ControlNet applications:

- Communications worksheet
- Device Tags worksheet

Appendix A contains a blank copy of each worksheet.

These points summarize the process of planning your DH+ or ControlNet communications using the worksheets:

- Identify each programmable controller the application will communicate with by defining nodes on the Communications worksheet.
- Define the foreground and background rates for the 11 scan classes.
- Define the programmable controller addresses for each object on your screen using the Device Tags worksheet. For each object on a screen, enter the object reference number, object type, and screen number as well as the tag definition. Use a new Device Tags worksheet for each screen.
- Record the addresses for your windows and PLC-controlled options on a different worksheet.

The Communications Worksheet

Record node and scan class information on the Communications worksheet.

Communications Worksheet

Application File Name: _____

PanelView Terminal Network Address: _____

Baud Rate: 57.6K (10,000 Ft)

Node Definitions

Node Name	PLC Type	Station Address	Timeout (sec)	Enabled
Unsolicited_Msgs (DH+)	PLC–2	*	3.000	

*The Unsolicited Messages Node is the PanelView terminal. By default, the terminal's Station Address is the DH+ station number configured in the current application. The station address can be changed in the PanelView terminal's Terminal Network Setup screen. This node is used in DH+ applications only.

Note: For more node definitions, use additional copies of this worksheet.

Scan Class Definitions

Name	Foreground (sec)	Background (sec)	Name	Foreground (sec)	Background (sec)

Note: The default foreground and background rates are 0.000 seconds and 5.000 seconds, respectively.

Defining Nodes

A node is a programmable controller that your application will be communicating with on a network. For each node, you must identify the type of programmable controller, its station address, and the timeout value.

Recording Node Information

Node Name	PLC Type	Station Address	Timeout (sec)
Unsolicited_Msgs	PLC-2	0	3.000

For each node, fill in:

- **Node Name:** Name assigned to the node. The name can be up to 40 characters long, and can include letters, numbers, the dash (-), and the underscore character (_). Spaces are not permitted.

  Tip: Use a name that will help you to identify your programmable controllers when you are assigning tags later in your application. For example, you could name your programmable controllers according to their locations or the machine or part of the process they will be controlling or monitoring.

DH+

Note: In DH+ applications, the node name “Unsolicited_Msgs” is reserved for system use in DH+ applications. It cannot be assigned to a node by the user. You cannot use this node name for ControlNet applications.

- **PLC Type:** The type of programmable controller.

DH+

For DH+ applications, specify one of the following:

- PLC-2, via DH+/DH Bridge
- PLC-3, via DH+/DH Bridge
- PLC-5, which includes the PLC-5/10, PLC-5/12, PLC-5/15, and PLC-5/25
- PLC-5 (Enhanced), which includes the PLC-5/11, PLC-5/20, PLC-5/30, PLC-5/40, PLC-5/60 and PLC-5/80
- SLC 5, which includes the fixed I/O SLC 5, SLC 5/01, SLC 5/02 and SLC 5/03 (OS300) (Direct DH+)
- SLC 5 (Enhanced), which includes the SLC 5/03 (OS301) and SLC 5/04 (Direct DH+)

ControlNet

For ControlNet applications, specify PLC-5 Enhanced, to use the PLC-5/20C and PLC-5/40C.

- **Station Address:** The address that identifies the programmable controller’s location on the network.

DH+

For DH+ applications, if the node is on the same highway as the PanelView terminal, it is a local node. If the node is on a bridged highway, it is a remote node. For details on determining the station address, see "DH+ Addressing," below.

ControlNet

For ControlNet applications, valid station addresses are between 1 and 99 (decimal).

Note: The PanelView terminal communicates with local ControlNet stations only. No ControlNet bridge devices are currently available to support remote communication.

Important: For versions 1.0 and 1.25 of ControlNet, the PanelView Terminal Network Address should not be set to 1.

- **Timeout:** The number of seconds the PanelView terminal should wait before reporting a communication error. Three seconds is usually enough and is the default when a new node is created.

DH+ and ControlNet

- **Enabled:** For DH+ and ControlNet applications, if a node is enabled, the terminal will communicate with the device at the node. If the node is not enabled, the terminal will not communicate with the device. Instead, the PanelView terminal stores values to be written to the device. This allows system designers to test their applications without setting up communications, and to avoid communication errors at the terminal during testing.

DH+ Addressing

Local Station Addressing: For a node connected to the same DH+ network as the PanelView terminal, valid station numbers are between 0 and 77 octal.

Remote Station Addressing: A PanelView terminal can communicate with nodes on remote DH+ networks. Remote DH+ networks can be bridged by a 1785-KA interface module or by a Pyramid Integrator that has an RM or KA module.

DH+ Networks Bridged by a 1785-KA

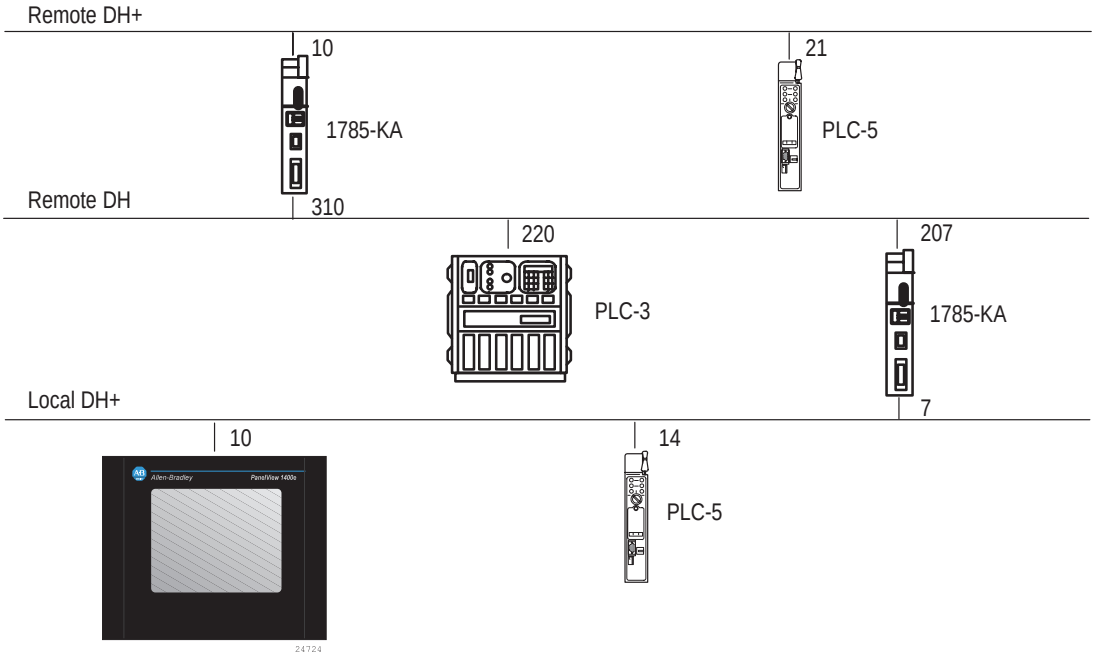
The format of the station address for nodes on remote DH+ networks is:

Station Address		Valid Entries
ll.rrr.ss		
ll	local bridge address	0 – 77 (octal)
rrr	remote bridge address	0 – 376 (octal)
ss	station address on remote DH+	0 – 77 (octal)
	or	
You can shorten the offlink address by combining the remote bridge and remote station addresses into one three-digit number. The system reads only the first digit of the remote bridge address, and adds the two-digit remote station address to produce a shorter address that still points to the same node.		
ll.rss		
ll	local bridge address	0 – 77 (octal)
rss	combined remote bridge address and remote station address	0 – 376 (octal)

The following figure shows a PanelView 1400e terminal connected to a DH+ network. The local DH+ network is bridged to a remote DH network, which is bridged to a second remote DH+ network. The 1785-KA interface modules are used to bridge the networks.

The valid station entries for the various nodes in the figure are:

Station Address	Valid Entries
14	14
21	7.310.21 or 7.321



PanelView 1400e Terminal

Multiple DH+ Networks Connected by a Pyramid Integrator

The format of the station address for nodes on a remote DH+ network is:

Station Address		Valid Entries
ll:rrrrr.ss		
ll	local bridge address	0 – 77 (octal)
rrrrr	PI Link number	0 – 65535 (decimal)
ss	station address on remote DH+	0 – 77 (octal)

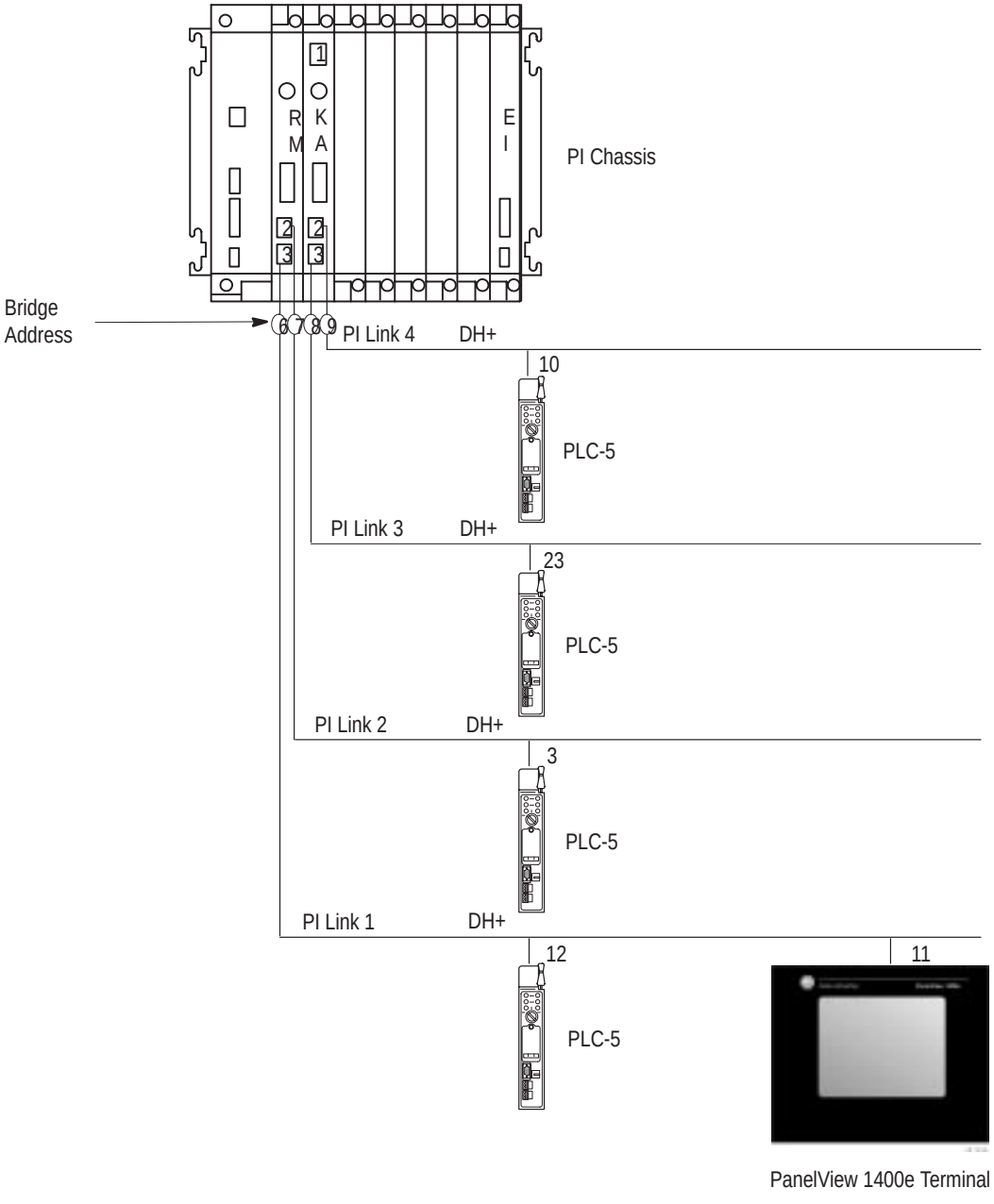
The Pyramid Integrator (PI) Link number for each DH+ network must be unique. Use the A-B 6200 Series Programming Software to configure link numbers. All station routing addresses must be local addresses.

A PI chassis contains one RM module with two network channels. It can also contain up to four KA modules, each with two network channels. Using the RM and four KA modules, a PanelView terminal can communicate with programmable controllers on any or all of 10 DH+ networks.

The figure on the following page shows a PanelView 1400e terminal connected to a DH+ network. The PI routes requests among four DH+ networks.

The valid station entries for the various nodes in the figure are:

Station Address	Valid Entries
12	12
3	6:2.3
23	6:3.23
10	6:4.10



Defining Scan Classes

When your application is running, it must periodically read values from the PLC controllers by scanning the PLC addresses.

You must assign all tags used in a DH+ or ControlNet application to a scan class. You can then assign each tag a different foreground and background scanning period (rate). The scan rate determines how often a tag value is updated.

A foreground period applies to the current screen, the alarm and information window addresses, and to the PLC-controlled PLC I/O Control options. The background scan period applies to all other screens in the system.

If a tag is used by both a foreground and a background component, it is scanned at the foreground rate.

Guidelines for Configuring Scan Classes

- Specify a frequent scan rate for tags representing fast-moving devices and a less frequent scan rate for tags representing devices that change less often.

  This approach optimizes system performance, providing high-speed scanning where required and saving on system resources by using less frequent scanning where acceptable.
- Set the scan period to be faster than the expected rate of change in the tag's data. For example, if you scan every five seconds for a change of state that might occur once an hour, the entire system is needlessly burdened. Conversely, if you scan too slowly, the system can't monitor value changes that might occur between scans.
- A scan period of zero means tags are scanned as quickly as possible. You can set foreground scan periods to zero, but never set background scan periods to zero. For foreground periods, use zero sparingly, because this setting can result in so much traffic that overall system communication slows down.
- Set background scan periods slower than foreground scan periods. Even if a screen isn't displayed, the tags associated with the screen are still scanned at the background scan rate. So keep the background rates infrequent enough so as not to slow down overall communication on the highway. A background period of 60 minutes is not too slow, because the necessary tags will be put on scan at the foreground rate when a new screen is displayed.

Name	Foreground	Background

Recording Scan Class Information

In the bottom part of the Communications worksheet, define how you want to configure each scan class:

- **Name:** The scan class name you want to assign to a particular node. By default, scan classes are named “A” through “K”. You can change these letters to any name up to 20 characters long. The name can include letters, numbers, and the underscore character. Spaces are not permitted.
- **Foreground:** The rate at which tags used in the current screen, in the Alarm and Information windows, in foreground and background trends, and for the PLC I/O control options are updated. Specify the speed in seconds or in fractions of a second.
- **Background:** The rate at which tags used in all screens not displayed are updated. Specify the speed in seconds or in fractions of a second.

Planning Addresses

After defining the screens for your application, you know the types and number of objects you will need. With the node and scan class definitions complete, you are ready to plan the object addresses.

For each screen, use one or more Device Tags worksheets to record tag information for each dynamic object. Also use the Device Tags worksheets to record information about Alarm and Information windows and PLC I/O Control options. These items are not specific to particular screens, so use a separate worksheet for them.

Guidelines for Organizing Tags to Optimize DH+ Network Performance

- Group tags as closely together as possible.
  - A consideration for optimizing network traffic relates to the addresses (PLC data file addresses) that are assigned to tags. PanelView reads or writes data in packets. A packet can contain up to 119 words of data (depending on the type of PLC). The data in a packet cannot cross data file boundaries. For example, the data cannot be from N7 and N10, only from one or the other.
  - When the PanelView terminal reads from the PLC, it tries to optimize the packet by fitting as much data into it as possible. If a tag is assigned the address N7:0 and another tag is assigned N7:118, the PanelView terminal will take all the data from N7:0 to N7:118 and fit it into one packet (depending on the type of PLC), using one read to get data for both tags.

- Group tags according to their scan classes.
  - If one of the tags was on the currently displayed screen, the packet would be scanned at the foreground rate of that tag. If both tags were on screens not currently displayed, the packet would scan at the fastest background rate assigned to those tags.
  - For example, if the tag assigned to N7:0 had a background scan rate of 2 seconds and the tag assigned to N7:118 had a background scan rate of 10 seconds, it will scan at the fastest rate of 2 seconds. As the packet contains all the data from N7:0 to N7:118, to optimize this packet you should assign tags to the addresses in between.
- Minimize the number of tag writes that must occur simultaneously
  - PanelView gives writes a greater priority than reads. Outstanding reads cannot be processed until all outstanding writes have been sent to the PLC and acknowledged. This means that display objects and tags may take longer than expected to update. This causes problems only when you have a busy network, combined with a large number of writes being generated in quick succession by the PanelView terminal.
  - A large number of writes can be generated under the following conditions:
    - assigning any or all of the Time and Date to PLC tags, as these tags write once every second.
    - when entering a screen, some objects write to their tag address. Problems can occur if there is a large number of write objects on one screen. Some objects that can cause problems are Scrolling Lists with large numbers of visible states assigned to them, and Numeric Input Cursor Points with many points defined.
    - when the application is first powered up, or is run for the first time, if default operation is not used in the Application Startup menu of the PanelView configuration screen. In this case you will write either the object's initial state value, or if it does not have an initial state it will write its initial value as defined in the tag.

Device Tags Worksheet

Application File Name: _____ **Screen Number:** _____

Object			Tag							Address		
Ref No.	Type	Initial State Value	Name	Type	Analog Tag Only: Scale	Analog Tag Only: Offset	Analog Tag Only: Data Type	Initial Value	Range of Values	Address	Number of Characters (string tag only)	Tag Start Value

Notes:
Initial State Value: Objects such as Multistate Push Buttons have a user-configurable initial state whose value will supersede the tag's initial value when the application is first run. This is called the tag's Start Value.
Tag Start Value: The start value is the tag initial value, or the object initial state value, if one exists.

For information about addressing syntax, refer to Appendix C, *Addressing Syntax.*

Object		
Ref No.	Type	Initial State Value

Recording Object Information

For each object, fill in the following:

- **Ref No.:** The object's unique identification number. Usually, you can start with 1 and increase by 1 for each object. Start from 1 again for each screen so that adding an object to the first screen doesn't require that all other numbers change.
- **Type:** The type of object, such as Interlocked Push Button.

  Some objects use several tags or expressions. See online Help or the *PanelBuilder 1400e Configuration Software for Windows Reference Manual*, or look at the PanelBuilder objects themselves, to plan which tags or expressions to use for each object.

  Use one line in the Device Tags worksheet for each tag and use the Type column to describe the function of the additional tags.

  Expressions are optionally used to read and manipulate the PLC output at the tag addresses. You can also use a line for each expression, to plan which ones you will need to define when you create the object. For information about expressions, see Chapter 7, *Creating Expressions.*
- **Initial State Value:** Objects such as Multistate Push Buttons have a user-configurable initial state whose value will supersede the tag's initial value when the application is first run.

Tag						
Name	Type	Analog Tag only			Initial Value	Range of Values
		Scale	Offset	Data Type		

Recording Tag and Expression Information

The tag database records all the addresses used throughout an application. It may also contain addresses the application doesn't use, for example, if the database is shared by another application. Instead of assigning an address directly to an object, you enter the address as a tag definition in the database, and then assign the tag, or an expression that uses one or more tags, to an object. If the same address is used by more than one object, you need only define it once in the database. For information about expressions, see Chapter 7, *Creating Expressions*.

Some objects require more than one control tag or expression. For example, a Maintained Push Button needs a Button Control tag and an Indicator State tag or expression. Record all necessary tag names and addresses in the worksheet, as well as information about the expressions.

For each expression, fill in the name of the expression in the object (for example, Indicator State), the name of the tag or tags the expression will read data from, and the manipulation the expression will perform on the tag data. For information about the kinds of operations expressions can perform, see Chapter 7, *Creating Expressions*.

For each tag, fill in the following:

- **Name:** A symbolic name you assign to this address. The tag name can be up to 40 characters long, and can include letters, numbers, the dash (-), and the underscore character (_). Spaces and periods (.) are not permitted.

  When you define tags in the Tag Database editor, you can also define folders. Folders are a convenient way to organize tags. For example, you may put all the tags for one screen's objects in one folder. The folder name becomes part of the tag name, and is separated from the rest of the name with a backslash. For example, the full name for the tag "StartButton_Control" in the folder called "Screen_1" is "Screen_1\StartButton_Control".

  For details on creating tags and folders, see Chapter 6, *Defining Tags*.

- **Type:** Specify one of these types of tags:

Tag Type	Type of Data Stored
Digital	A single bit, giving the values 0 or 1. Use a digital tag for devices that can only be on or off, such as push buttons.
Analog	Range of values (depends on the data type selected). Use an analog tag to control or monitor a value, for example, a temperature or the position of a rotary control. When you define an analog tag, you must also fill in the Scale, Offset and Data Type fields.
String	ASCII string, series of characters or whole words (maximum of 82 characters). Use a string tag only with the ASCII Input and ASCII Display objects, and (optionally) for sending alarm messages to the PLC.
Block	A contiguous bit string of up to 1024 bits. A block tag is used strictly for defining the Alarm Triggers and Acknowledge to PLC controls. These controls can monitor and acknowledge multiple simultaneously active alarm conditions. For details, see Chapter 11, *Configuring Alarms*. **Note:** If you assign a block tag in a DH+ or ControlNet application, the application will be compatible with PanelView Version 02.00.00 or later. It will not be compatible with earlier versions.

- **Scale (Analog Tag Only):** Specify a value, with 1 being the default. The scale and offset work together to modify any raw data that comes from or is written to a programmable controller, for example, to convert the raw data into engineering units.
- **Offset (Analog Tag Only):** Specify a value, with 0 being the default. The scale and offset work together to modify any raw data that comes from or is written to a programmable controller.

  When the PanelView terminal reads a value from the PLC, it multiplies this value by the scale value and then adds the offset before it writes the value to the terminal's value table. Similarly, when the terminal sends data to the PLC, it first subtracts the offset and then divides by the scale value to convert the value back to raw data before it is written to the PLC.

  With Scale = 1 and Offset = 0 (the default values), PLC data is not changed.

  ▶ **Note:** For PLC output from Analog and Digital tags, you can create expressions to manipulate or convert the data as needed.

- **Data Type (Analog Tag Only):** Unless you have specific reasons for doing so—that is, if you want a special manipulation of the data—use the Default data type.

  ▶ **Note:** Not all PLCs accept all data types. For the list of valid data types for the PLC you are using, see your PLC's User Manual.

An example of special manipulation of the data is to treat the data in the integer file N7:0 as a 4-digit BCD. In this case you would specify 4-BCD as the data type.

Important: No matter what data type is assigned, the tag's entire address is read when a tag value is retrieved from the programmable controller. For example, for a tag with the address N7:00 and byte data type, the terminal will still look at all 16 bits in the 0 element of integer file 7. Ensure that any unused bits in the address are 0 to prevent invalid values from being read. Note that writing to a tag will update all bits in the address, not just those implied by the tag's data type.

You can use the following:

- **Default:** If you select Default, the data type is dictated by the address. For example, if the address is N7:0 (an integer file), the data type would be integer.
- **Unsigned Integer:** An unsigned integer occupies one 16-bit word, stored in binary format. The range of raw data values is from 0 to +65535.
- **Integer:** A signed integer occupies one 16-bit word, stored in two's complement format. The range of raw data values is from –32,768 to +32,767.
- **Long Integer:** A long integer occupies two 16-bit words, stored in two's complement format. The range of raw data values is from –2,147,483,648 to +2,147,483,647.
- **Floating Point:** Single-precision (32-bit) floating point. The range of data values is –3.402823E +38 to –1.175494E –37, 0, and +1.175494E –37 to +3.402823E +38.
- **Byte:** The byte data type occupies 8 bits of unsigned data. The range of raw data values is from 0 to 255.
- **Binary:** You can use this data type for Remote I/O applications only.
- **Bit Position:** This data type is a contiguous bit string. The position of the least-significant bit in the bit string determines the value of the tag. For value ranges from 0 to 16, use a 16-bit address. For value ranges from 0 to 32, use a 32-bit address.

- **BCD (Binary Coded Decimal):** If you assign one of the BCD data types, the object's PLC control word will be a contiguous bit string of one of these numbers of bits:

This number of BCD digits	Requires this many bits
1 BCD digit	4
2 BCD digits	8
3 BCD digits	12
4 BCD digits	16
5 BCD digits	20
6 BCD digits	24
7 BCD digits	28
8 BCD digits	32

Note: If you assign 5 to 8 BCD digits the device must support Long Integer data.

Each group of four bits represents the equivalent of one decimal digit in the tag value. The number of bits required depends on the maximum value required for the object. This depends on the application.

Note: The BIN3, BIN4, BIN6, and BIN8 data types are reserved for use in Modbus applications. For more information, see the *PanelBuilder 1400e Modbus User Manual,* Publication Number 2711E-6.12, which is supplied as part of the optional Modbus Communications Kit, Catalog Number 2711E-UMOD.

- **Initial Value:** The value to be placed in the input address of an object upon power-up. Assign Initial Values to objects with PLC input addresses.

Note: For any multistate object you draw in the Screens editor, you can define an Initial State. If you define an Initial State for the object and an Initial Value for the object's control tag, the object's Initial State takes precedence. The tag's initial value is used only if an initial state is not defined.

- **Range of Values:** If an object has a range of state values, record the values in this column.

Range of Values

Recording the Address

For each tag, record the programmable controller address as follows:

Address

- **Address:** The tag address. Record the full tag address of digital and analog tags in this column. Except for string and block tags, the length of the address will be the size of the data file element in the address.

  For string and block tags, record the address of the first bit in the array in this column. The data file location in the PLC must accept string tag addresses. The address must start on an element boundary (sub-element starting addresses are not supported). The address cannot cross PLC data file boundaries.

  The syntax depends on the type of address or controller. See Appendix C, *Addressing Syntax*.

Number of Characters (string tag only)

- **Number of Characters (String Tags) or Number of Bits (Block Tags)**
  - For String tags, specify the length of the string in characters. The length of the string must be a multiple of the number of bytes in the address.
  - For Block tags, specify the number of bits in the array, up to 1024. The number of bits must correspond to the element in the address. For example, if the address is located at an integer file element, the length must be a multiple of 16, the bit length of an integer.
- **Tag Start Value:** The tag start value is the initial value, or the object's initial state value, if one exists.

  For block tags, the initial value can be 0 or 1.

Carrying Out Your Plan

After you complete the planning steps in this chapter, your worksheets contain all the information you need to create your application. To create your application, work through the remaining chapters in this manual, as described in "Steps for Building an Application," at the beginning of this chapter.

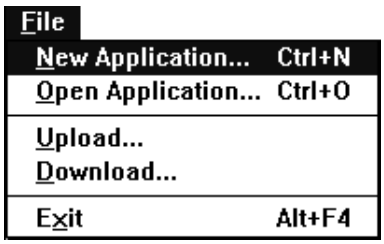
Working with Applications

This chapter introduces you to the basics of working with applications. It shows you how to:

- create new applications
- open existing applications
- convert applications created in PanelBuilder for DOS or PanelBuilder 1200 to PanelBuilder 1400e format
- save applications
- copy applications
- change the application name
- create application reports
- close applications
- delete applications

Important: This chapter assumes you have a good working knowledge of the Microsoft Windows operating system. For detailed information, refer to your *Microsoft Windows User's Guide* (if you use Windows 3.1), *Introducing Microsoft Windows 95* (if you use Windows 95), or *Introducing Microsoft Windows NT Workstation* (if you use Windows NT).

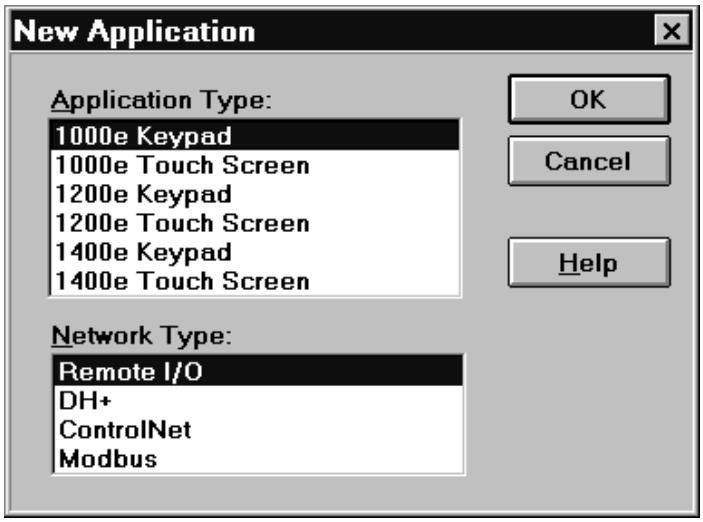
Creating a New Application



To create a new application:

1. Choose New Application from the File menu or from the toolbar, or press CTRL+N.

   The New Application dialog box appears.



2. Select the type of terminal you will be using to create this application. Choose one of the following terminal types:
   - PanelView 1000e Keypad
   - PanelView 1000e Touch Screen
   - PanelView 1200e Keypad
   - PanelView 1200e Touch Screen
   - PanelView 1400e Keypad
   - PanelView 1400e Touch Screen

   Important: You can't convert a keypad application to a touch screen application, or vice versa. However, you can switch between 1000e, 1200e, and 1400e application types. See "Changing the Application Type," in Chapter 12, *Configuring Terminal Setup Options*.

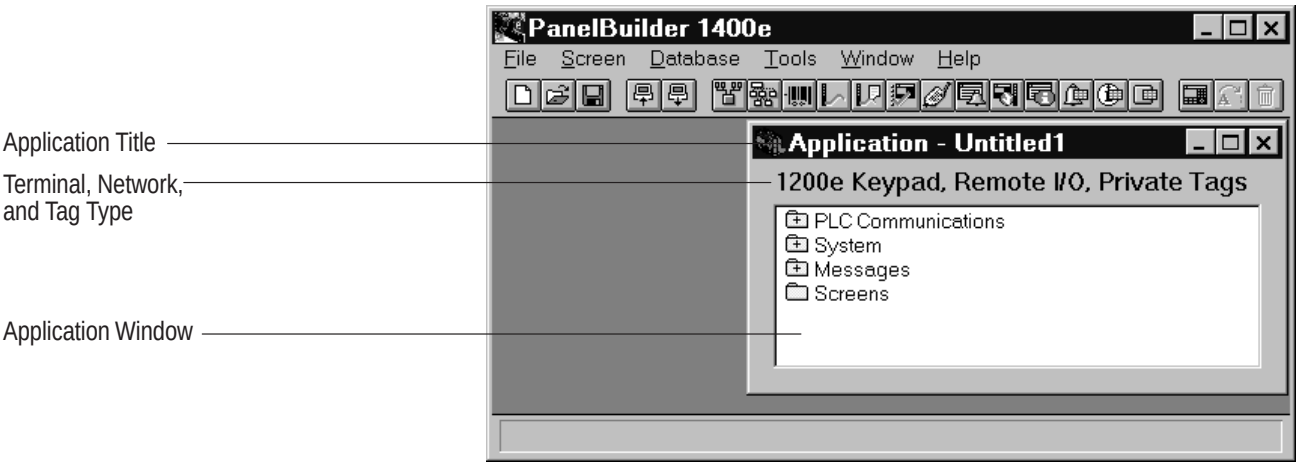
3. Select the network you will be using to create this application. Choose one of:
   - Remote I/O
   - DH+
   - ControlNet

Note: For information about creating a Modbus Application, see the *PanelBuilder 1400e Configuration Software for Windows Modbus User Manual*, Catalog Number 2711E-UMOD.

Note: You can convert an application from one communication network type to another. If you do change network types, you may lose node, scan class, rack, or block transfer definitions. See "Changing the Network Type" in Chapter 5, *Defining Communications*, for more information.

4. Choose OK.

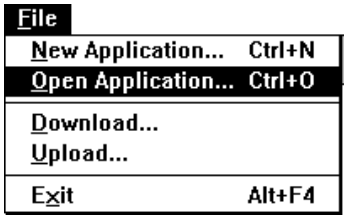
   The Application Window opens inside the PanelBuilder window. Note that the application is still untitled, and the PanelView terminal and network types are listed. You must assign a name when you save the application.



Opening Applications

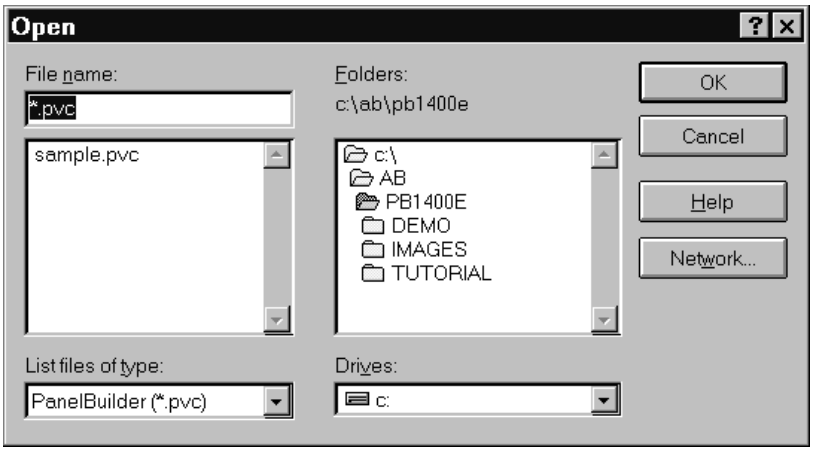
You can open an application created in PanelBuilder for DOS, PanelBuilder 1200, or PanelBuilder 1400e. For information on converting a PanelBuilder for DOS or PanelBuilder 1200 application, see the following section.

To open an existing application:



1. Choose Open Application from the File menu or from the toolbar, or press CTRL+O.

   The Open dialog box appears. It lists the available applications in the specified drive and directory. If no applications appear, you may need to change the drive or directory, or select a different file type in the List Files of Type drop-down list.



2. If you know the path and name of the application, you can type this in the File Name field. If you don't know the application name, select the type of application you want to display by making a selection in the List Files of Type drop-down list. Your choices are:
   - PanelBuilder 1400e configuration files (.pvc)—this is a configuration file created in PanelBuilder 1400e
   - PanelBuilder 1400e terminal files (.pvd)—this is a download file ready for downloading to a PanelView 1200e or 1400e terminal
   - PanelBuilder 1200 files (.pbw)—this is a configuration file created in PanelBuilder 1200
   - PanelBuilder for DOS files (.cfg)—this is a configured application file created in PanelBuilder for DOS
   - PanelBuilder 1200 terminal files (.apl)—this is a download file ready for downloading to a PanelView 1200 terminal
   - all files (*.*)
3. Choose the application you want to open.
4. Choose OK.

   The Application Window for the application you just selected appears on your screen. If you're opening a .cfg, .pbw, or .apl application, PanelBuilder automatically converts the application. A status window appears while the application is being converted.

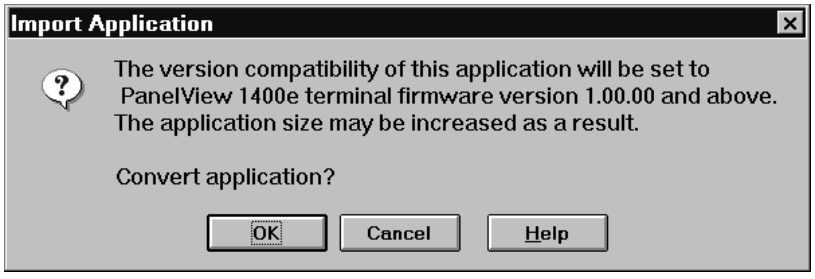
Converting Applications Created in PanelBuilder 1200 or PanelBuilder for DOS

To run your PanelBuilder 1200 or PanelBuilder for DOS applications on a PanelView 1000e, 1200e, or 1400e terminal, you must convert the application. You can convert any .cfg, .pbw or .apl files.

To convert a PanelBuilder 1200 or PanelBuilder for DOS application:

1. Open the application as described in the previous section.

   The following message appears.


Import Application
The version compatibility of this application will be set to PanelView 1400e terminal firmware version 1.00.00 and above. The application size may be increased as a result.
Convert application?
OK
Cancel
Help

2. Choose OK.

A progress window appears during the time it takes to convert the application.

Once the application is open, you can edit it. This application is now compatible with PanelBuilder 1400e only, and will run on PanelView 1000e, 1200e, and 1400e terminals only. However, the original file remains unchanged on the computer's disk. When you save or close the converted file, it will be saved as a .pvc file. The .pvc file format is the PanelBuilder 1400e configuration file format.

The Conversion Process

When the PanelBuilder for DOS or PanelBuilder 1200 application file is converted, the following changes are made:

- a tag database is generated, and a tag is defined for each unique address that had been associated with objects, windows, and Global Address options
- the rack and block transfer definitions are generated
- each object, window, and Global Address option that used to have an address associated with it now has the correct tag assigned to it
- all screen components are converted to their pixel graphic equivalent. For example, two connected lines will now appear as two regular lines; buttons will have the new three-dimensional button borders, and so on
- lines of characters are converted to text
- the file format is .pvc

When you open an application created with PanelBuilder for DOS, PanelBuilder 1200, or an earlier version of PanelBuilder 1400e than Version 4, the following Version 4 default values are set:

- the application's .pvc file is saved as a PanelBuilder Version 4 file. It is set to run on a PanelView terminal using firmware Version 1 or later. To change the Target Firmware Version, see "Specifying Target Firmware Version," next.
- the Message to PLC option for Alarm Messages is set to No
- the Retain Last List State and Skip State options for Cursor Lists are set to No (in Scrolling List objects)
- if the application uses an Enhanced SLC device, the Slot Number field in the Network File Transfer tab of the Configure Communication Setup dialog box is set to 0
- if the application has a block tag assigned to the Alarm Triggers control, the tag uses the Bit Alarm Trigger Type with a Starting Trigger of 1; if the Alarm Triggers control is an analog or digital tag, it uses the Word Alarm Trigger Type

- if the application has a block tag assigned to the Acknowledge to PLC control, the tag uses the Bit Acknowledge to PLC Type with a Starting Trigger of 1; if the Acknowledge to PLC control is an analog or digital tag, it uses the Word Acknowledge to PLC Type
- if the application has messages that contain the character sequence "\n" the sequence is converted to "\\n" to prevent a validation error. The sequence will appear as "\n" when you download the application to the PanelView terminal.

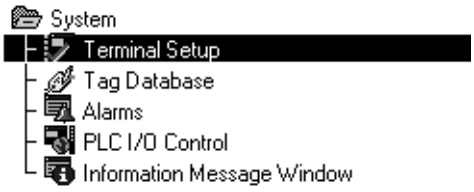
See Appendix E*, Version Control*, for a list of features that make applications incompatible with earlier versions of the PanelView firmware.

Specifying Target Firmware Version

You can select the version of PanelView firmware that you plan to use when you run your application. The Target Firmware Version you choose allows your application to run on the listed version, as well as later versions of the firmware:

- **Version 1 and up**
- **Version 1.03 and up** allows your application to run on a PanelView terminal connected to the ControlNet communication network
- **Version 2 and up** allows you to use the new features that come with PanelBuilder, Version 2
- **Version 3 and up** allows you to use the new features that come with PanelBuilder, Version 3
- **Version 4 and up** allows you to use the new features that come with PanelBuilder, Version 4

When you validate your application, PanelBuilder warns you if a feature you have used will make the application incompatible with the Target Firmware Version. See Appendix E*, Version Control*, for a list of features that make applications incompatible with earlier versions of the PanelView firmware.



To specify the Target Firmware Version:

1. Open the System folder and choose Terminal Setup, or choose from the toolbar.

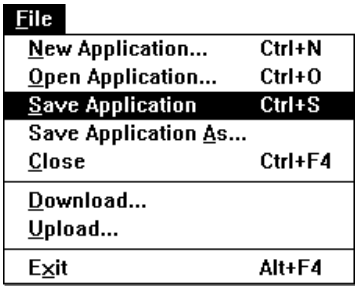
   The Configure Terminal Setup dialog box appears.

2. In the Target Firmware Version field, select a version from the drop-down list.
3. Choose OK.

Saving Applications

When you save an application, you save all work that you created in the editors.

▶ **Tip:** To avoid losing work due to a system failure, save your application immediately after you've made important or extensive changes.



There are two ways to save an application:

- **Save Application** overwrites the originally opened version of the application.
- **Save Application As** saves an open application with a new name, in a new directory, or in another file format.

To save changes to an open application:

▶ Choose Save Application from the File menu or from the toolbar, or press CTRL+S.

PanelBuilder replaces the previous version of the application with the new version.

Tip: You can reclaim the disk space occupied by deleted parts of your application, by compressing your application file. When you make extensive edits to an application, especially if you delete existing screens or remove bitmaps from the library, PanelBuilder doesn't free the space those objects or screens occupy in your application. This makes your files quicker to save, but requires more disk space in which to save them.

To compress an open file:

1. Choose Save Application As from the File menu.
2. Check the Compress field.
3. Choose OK.

To save an open application under a new name or in another directory:

1. Choose Save Application As from the File menu.

   The Save As dialog box appears. The application name is the default file name.
2. Type the new name for your application using a maximum of eight characters, or change the default drive or directory.

Note: When used with Windows 95 or Windows NT, PanelBuilder does not support long file and directory names. Ensure that the file and directory names you use conform to the DOS naming standard. If you enter an invalid file or directory name, an error message will appear, and you will be prompted to re-enter the file name.

3. Choose OK.

Saving an Application in Another File Format

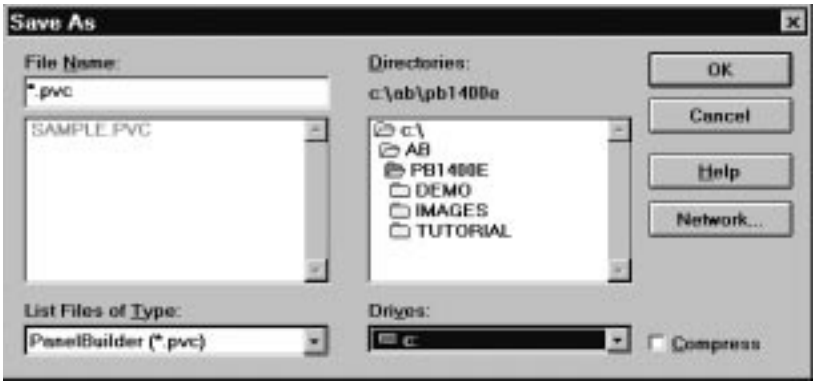
Save your applications as:

- **.pvc**—The file format for PanelBuilder 1400e. This is the default format.
- **.pvd**—The file format for an application to be downloaded to a PanelView terminal.

To save an application as another file type:

1. Choose Save Application As from the File menu.

   The Save As dialog box appears. If you're saving a new application, the file name field displays *.pvc. If you're saving an existing application, the application name is the default file name.



2. Enter the new name for your application using a maximum of eight characters, or change the default drive or directory. If you're saving the application as a different type of file, type the appropriate file extension. The application will be saved as the specified file type.

   Important: The .pvc and .pvd file extensions determine the file type the application will be saved as. If you don't type a file extension, the application will be saved according to the file type currently selected in the List Files of Type field.

▶ **Note:** When used with Windows 95, PanelBuilder does not support long file and directory names. Ensure that the file and directory names you use conform to the DOS naming standard. If you enter an invalid file or directory name, an error message will appear, and you will be prompted to re-enter the file name.

3. To minimize the file size, check the Compress field.

   The file size doesn't shrink when you edit and save the file, even if you've deleted screens or removed bitmap images from the library. Compressing saves fragmented .pvc files in as small a size as possible. Compress has no effect on .pvd files.

▶ **Note:** For large applications with many screens, compressing the file can take a few minutes.

4. Choose OK.

Copying Applications

To copy an application, you can use the Save As command, and save the application under another name. See the previous section for more information.

You can also use the Windows Explorer in Microsoft Windows to copy the application. The application is stored in a file of the same name as the application, with a .pvc file extension. See your *Microsoft Windows User's Guide* for details on how to use Windows Explorer.

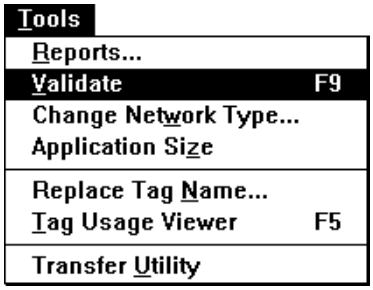
Tip: To make a copy of an application with a shared tag database to move to another machine, first convert the database from Shared to Private, then copy the application. Then only a single .pvc file will need to be moved.

Renaming Applications

You can use the Save As option described previously to rename an existing application, or you can use the Windows Explorer. For details on using the Windows Explorer, see the *Microsoft Windows User's Guide.*

Validating Applications

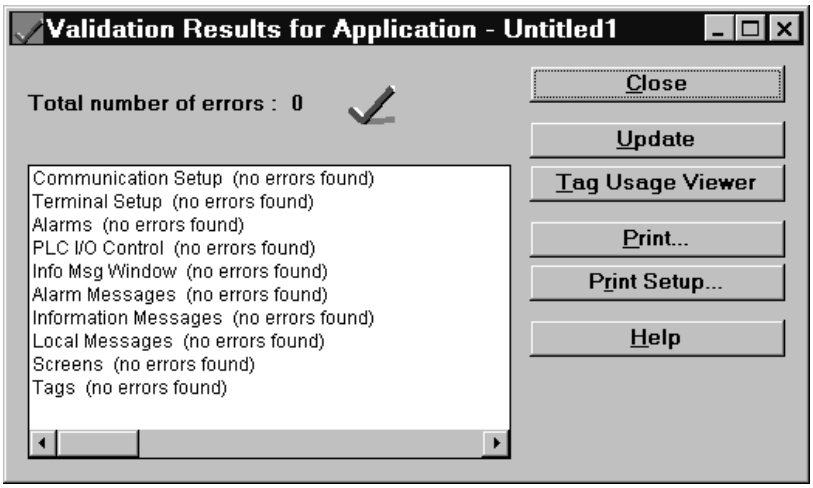
As you create your application, periodically validate it to ensure there are no problems. The Validate option searches for application errors in Terminal Setup, Alarms, PLC I/O Control, Information Message Window, the message editors, screens, tags, and expressions. The Validate Option locates errors such as overlapped input screen objects, offscreen objects, conflicting tag initial values, invalid messages, invalid expression syntax, and version control violations.



To validate your application:

1. Choose Validate from the Tools menu, or press F9.

   A progress bar is displayed, then the Validation Results window appears. It lists and describes the errors. If there are no errors, a check mark appears (✓).



2. If there are errors listed, go back to the dialog box, screen, or message editor and correct the errors.

   Tip: Use the Tag Usage Viewer to help understand any tag-related errors.

Determining the Application Size

PanelBuilder calculates the size of the application (the size of the .pvd file).

The application's maximum allowable size depends on the type of terminal it will run on and whether the application will be stored on the terminal's memory or on the PCMCIA memory card. To ensure your application will fit on its destination, determine the capacity of the space where it will be stored, and check the application's size before downloading. The maximum allowable application size, regardless of the file's destination, is 8 MB.

- On a PanelView 1200 Series F or G terminal, the application and firmware both reside on the 4 MB PCMCIA memory card that comes with the PanelView 1200 Enhancement Kit (Cat. No. 2711E-U1B12C). There will be approximately 2.2 MB available for the application.
- On a PanelView 1200e or 1400e terminal, the application can reside in either Flash memory (if the application is less than 256K) or on the PCMCIA memory card (if the application is more than 256K).

- On a PanelView 1000e terminal, the application can reside in either Flash memory (if the application is less than 2,256K) or on the PCMCIA memory card (if the application is up to 15.75 MB).

To determine the application size:

▶ Choose Application Size from the Tools menu.

After processing, a message appears indicating the size of the application.

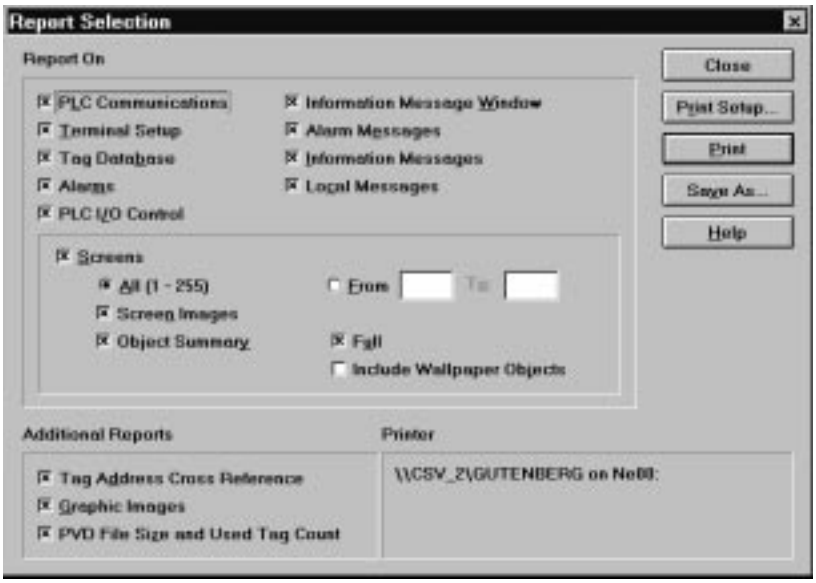
Creating Application Reports

Application reports help you document and track the screens, objects, tags, expressions, and messages. You can print these reports or save them as ASCII text files.

Before you create an application report, save the application file. Then make the report selections.

To open the Report Selection dialog box:

1. Open the application you want to document.
2. Choose Reports from the Tools menu.



Report On Options

Because a full application report can be quite large, you may want to document only certain features of the application. Choose the application features you want to document in the Report Selection dialog box. The options are described in the table below.

Note: Some ASCII symbols may appear as their numeric equivalents on the report. For example, the ALT+009 symbol appears as \009 on the report. The symbols will appear as expected when the text is displayed on the PanelView terminal.

Note: The features you select are saved in the PB1400E.INI file in the Windows directory for the next time you create a report.

This report option	Does this
PLC Communications	• for a DH+ or ControlNet application, shows the communications setup (network type, the terminal's network address and PanelView Location Routing Address, and network download control tags) and node and scan class definitions. • for a Remote I/O application, shows the communications setup (network type, baud rate, PanelView Location Routing Address, and network file transfer information) and rack and block transfer file definitions.
Terminal Setup	Shows the Application Startup screen number, application file comment, application type, Target Firmware Version, the timing parameters, and whether the Retain Cursor on Cancel option is enabled or disabled.
Tag Database	Lists alphabetically all the tags in the database that are used in the application and shows each tag's definition.
Alarms	Shows the configuration of the Alarm Window, Alarm History, and Alarm Status screens and their associated tags or expressions.
PLC I/O Control	Lists all the PLC I/O Control options and identifies the tags or expressions that have been assigned to these options.
Information Message Window	Shows the configuration settings for the Information Message Window and its tags or expressions.
Alarm Messages	Lists all the messages, including trigger values, created in the Alarm Message editor, and their options.
Information Messages	Lists all the messages, including trigger values, created in the Information Message editor.
Local Messages	Lists all the messages, including trigger values, created in the Local Message editor.
Tag Address Cross Reference	Lists all the places in the application where tags are assigned, sorted by tag address.
Graphic Images	Lists the bitmap (*.bmp) and AutoCAD graphics (*.dxf) files imported into the application. Note that PanelBuilder saves .dxf images as .wmf files once imported, so .dxf files appear as .wmf files on the report.
PVD File Size and Used Tag Count	Lists the .pvd file size and the number of tags used in the application.

To select Report On options:

1. Select the fields of the features you want to document.

   An X appears in the fields of the features selected for the report.

2. To deselect an option, click the field so the X disappears.
3. When you've selected all the options you want to document, choose Print to print the report.
4. When you're finished, choose Close to close the dialog box.

Report On Screens Options

By default, all the application screens are selected for printing. If you don't want to print all the screens, you can select a range of screens for printing. You can also select from a range of screen print options. These are described in detail in the table below.

To select a range of screens:

1. With Screens selected, select the From field in the Screens section.
2. Type the number of the first screen you want to print.
3. Select the To field and type the number of the last screen you want to print.

The options for Screen Reports are described in the following table:

This report option	Does this
Screen Images	Shows the image of each screen and all its objects. The Screen Images report is available only when you print the report to a printer, not when you save the report to a file.
Object Summary	Lists screen objects and associated information for the screens you've selected. You can choose two options for this report: • Full • Include Wallpaper Objects If you don't choose one of these options, you get a brief description of all the non-wallpaper objects.
Full	Provides detailed configuration information on all objects.
Include Wallpaper Objects	Includes all drawing objects that have been converted to wallpaper.

Print Setup

How you can set up the printer depends on how your printer is configured in Microsoft Windows. For details on how to set up your printer, refer to your *Microsoft Windows User's Guide.*

You can use any printer supported by Windows to print reports.

To set up the printer:

1. Choose Reports from the Tools menu.
2. Choose Print Setup in the Report Selection dialog box.

   The Print Setup dialog box appears.
3. Select the printer, and desired page orientation and size.
4. Choose OK.

See online Help for information about the Options and Network buttons.

Solving Printing Problems

If you can't print, follow these steps:

1. Check your printer's manual to ensure you've assigned the correct printer port setting, and that the port is configured correctly.
2. Ensure your printer is turned on, is online, and supplied with paper.
3. Ensure you've selected report options.

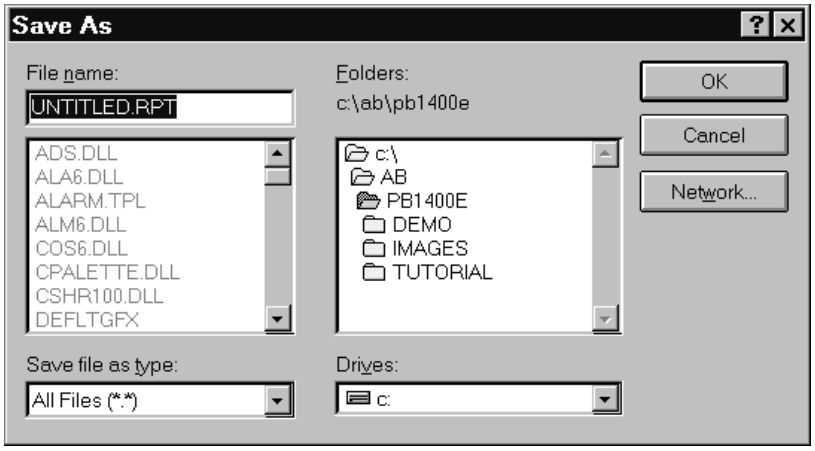
Saving the Report as an ASCII Text File

You can save the application report as an ASCII text file.

To save the report as an ASCII text file:

1. Choose the Save As button in the Report Selection dialog box.

   The Save As dialog box appears.

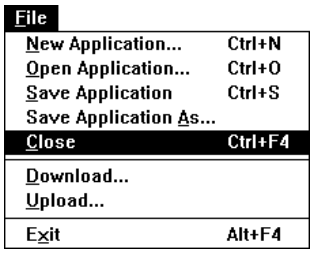


   By default, the file name is the same as the application name. The file extension is .rpt.

2. To change the file name, type the new file name in the File Name field.
3. Change the drive or directory, if required.
4. Choose OK.

Note: The ASCII text file does not include screen images.

Closing Applications



When you close an application, all open editors for that application close as well.

To close an application:

1. Choose Close from the File menu or from the Control menu; press CTRL+F4; or double-click the control menu box on the Application Window.

   If you made any changes to the application since last saving it, PanelBuilder prompts you to save the changes.

2. Choose Yes or No.

   Important: If you choose No, you lose changes you made in the configuration dialog boxes or editors. However, if you have a shared database, those changes will already have been written to disk.

Deleting Applications

Use the Windows Explorer to delete an application. The application is stored in a file of the same name as the application, with a .pvc file extension.

See your *Microsoft Windows User's Guide* for details on how to use the Windows Explorer.

Working with Editors

PanelBuilder has four types of editors:

- dialog boxes
- form/spreadsheet editors
- tag database editor
- screen editor

This chapter covers the dialog boxes and form or spreadsheet editors. Instructions for using the tag database and screen editors are in Chapter 6, *Defining Tags*, and Chapter 8, *Creating Screens*.

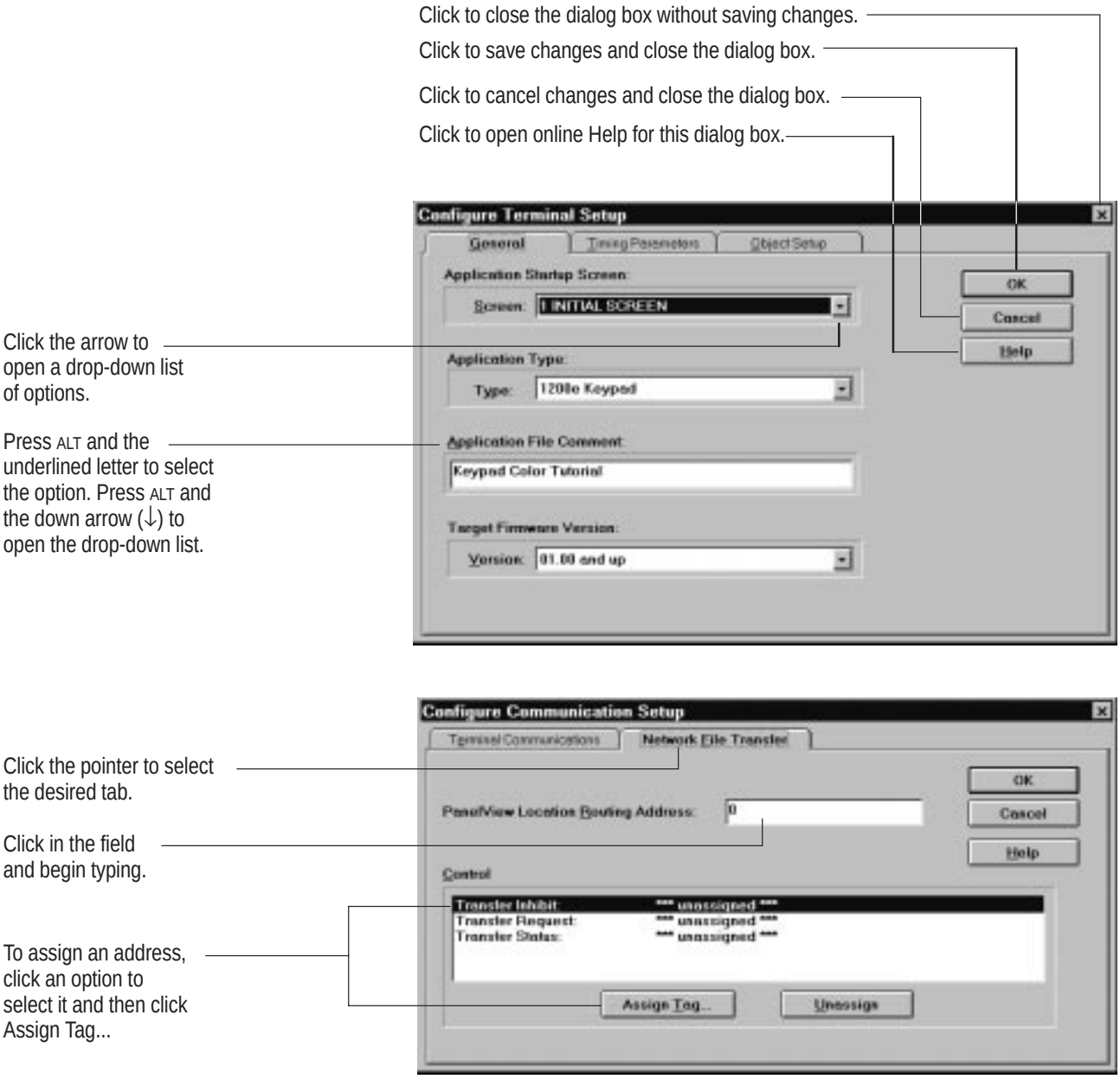
Dialog Box Editors

Five of the PanelBuilder editors are dialog boxes:

- Configure Communication Setup
- Configure Terminal Setup
- Configure Alarms
- PLC I/O Control
- Configure Information Message Window

Note: Many of the PanelBuilder screen objects have Configuration dialog boxes. The instructions here also apply to the object Configuration dialog boxes. See “Configuring Objects” in Chapter 9, *Creating Objects,* for more information.

The Configure Communication Setup dialog box, shown next, contains many of the features of these editors. For more information on how to navigate through a dialog box, see your Windows documentation.

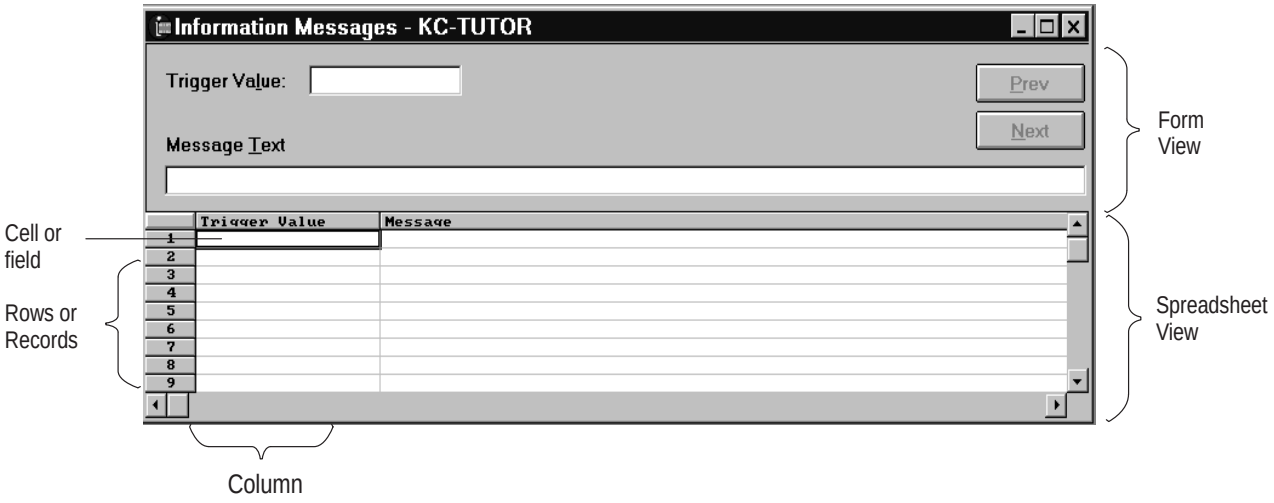

Click to close the dialog box without saving changes.
Click to save changes and close the dialog box.
Click to cancel changes and close the dialog box.
Click to open online Help for this dialog box.
Click the arrow to open a drop-down list of options.
Press ALT and the underlined letter to select the option. Press ALT and the down arrow (↓) to open the drop-down list.
Configure Terminal Setup
General
Timing Parameters
Object Setup
Application Startup Screen:
Screen:
1 INITIAL SCREEN
OK
Cancel
Help
Application Type:
Type:
1208e Keypad
Application File Comment:
Keypad Color Tutorial
Target Firmware Version:
Version:
01.00 and up
Click the pointer to select the desired tab.
Click in the field and begin typing.
To assign an address, click an option to select it and then click Assign Tag...
Configure Communication Setup
Terminal Communications
Network File Transfer
PanelView Location Routing Address:
0
OK
Cancel
Help
Control
Transfer Inhibit:
Transfer Request:
Transfer Status:
*** unassigned ***
Assign Tag...
Unassign

Form/Spreadsheet Editors

The Nodes, RIO Racks, and RIO Block Transfer Files editors and the three Messages editors (Information Messages, Alarm Messages, and Local Messages) all have a form and a spreadsheet component.

Using the Spreadsheet and Form

A spreadsheet is a grid of rows and columns. Each intersection of a row and a column forms a cell.

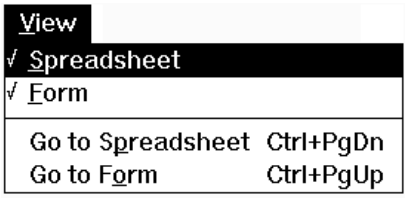


Before you begin working in the spreadsheet, familiarize yourself with these terms:

- **Record**—Each row in the spreadsheet is called a record. A record is one complete entry. For example, in the Node editor, one record is one complete definition of a PLC controller. The form area displays one record at a time.
- **Field**—Each cell in a row constitutes a field.

You can view information in the form, in the spreadsheet, or both at once. The default view is the spreadsheet for the Messages editors, and both for the Node, Rack, and Block editors.

The View menu option with a check mark (✓) is the selected view. If both Spreadsheet and Form have check marks, both views are selected. If only one view is selected, the displayed view is dimmed so that it cannot be deselected.



To view both the spreadsheet and the form:

▶ Ensure both the Form and Spreadsheet options in the View menu have check marks. If an option doesn't have a check mark, choose it.

Both the spreadsheet and the form appear.

To view only the spreadsheet:

▶ If Form has a check mark and Spreadsheet doesn't, you cannot deselect Form. Choose Spreadsheet from the View menu. Then choose Form to deselect it.

If both have check marks, choose Form so that its check mark disappears.

The spreadsheet appears. You can create or edit records in this area when you use the Messages editors.

To view only the form:

▶ If the Spreadsheet option has a check mark and Form doesn't, you cannot deselect Spreadsheet. Choose Form from the View menu. Then choose Spreadsheet to deselect it.

If both have check marks, choose Spreadsheet so that its check mark disappears.

The form for the currently selected record appears. You can create or edit records in this area.

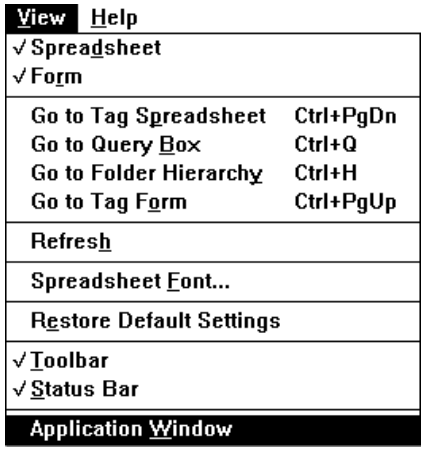
Moving between the Spreadsheet and the Form

When the spreadsheet and the form are open at the same time, you can move from one to the other.

- If you're using the mouse, click in any field of the spreadsheet or the form. The cursor will appear, and you'll be able to enter text.
- If you're using only the keyboard, choose Go to Spreadsheet or Go to Form from the View menu, or press CTRL+PGDN or CTRL+PGUP.

Using Multiple Editors

You can have multiple Messages editors open at one time to copy text from one editor to another, or to view messages in different editors at the same time.



To open multiple Messages editors:

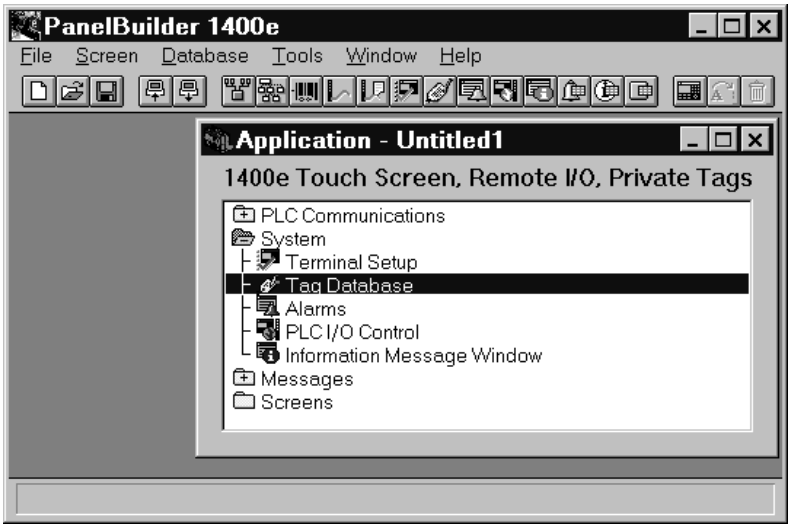
- With one Messages editor open, double-click another Messages editor in the Application window.

You can also open the Tag Database and Node editors at the same time for DH+ and ControlNet applications. For Remote I/O applications, you can open the Tag Database, Rack, and Block editors at the same time. To do this, use the Application Window.

To use the Application Window:

- With one of the editors open, click or choose Go to Project Manager from the View menu.

  The Project Manager dialog box opens.

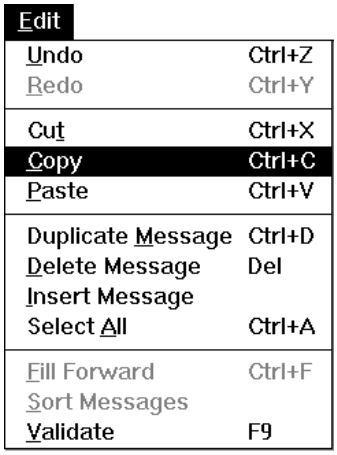


Working in the Spreadsheet

In Form view, you can edit only one record at a time, using the Prev (Previous) and Next buttons to scroll through the records.

In Spreadsheet view, in the Messages editors, you can edit many records at once, as described in "Editing Selections," later in this chapter. In the Node, Rack, and Block editors, you can click on a row in the spreadsheet to edit the record in the form. For the Node, Rack, and Block editors, you can't type entries directly into the spreadsheet.

Note: The editors do not all operate in the same way. The three Messages editors can do all the editing described in the remainder of this chapter. The commands with an asterisk (*) in the table below are commands you can use in the Node, Rack, or Block Editors.



You can edit records using the tools from the toolbar, or commands from the Edit menu.

This menu command	Or this toolbar icon	Does this
Undo		Undoes the most recent action or series of actions.
Redo		Redoes actions that you undid using the Undo command.
Cut		Cuts the selected block of cells, records, or columns.
Copy		Copies the selected block of cells, records, or columns.
Paste		Pastes the previously cut or copied block of cells, records, or columns into a screen area.
Duplicate *		Duplicates the selected record or block of records.
Delete *		Deletes the selected record or block of records.
Insert *		Inserts a blank record between existing records.
Fill Forward		Copies identical data forward into the selected area.
Sort ** or Sort Messages		Arranges the records alphabetically or numerically.

* supported in the Node, Rack, and Block editors
** supported in the Rack and Block editors

Using the Undo and Redo Commands (Messages Only)

Before you begin creating or editing records, you should know how to use the Undo and Redo commands. Use these commands any time you create or edit a record and then decide that you want to reverse your actions. The Sort command is the only operation you can't undo.

When you undo an operation and then change your mind, you can reverse your action using the Redo command. You can use the Redo command as many times as you used the Undo command. When you move the pointer over the Undo or Redo icons, the status bar displays the action that will be performed.

Note: You cannot use Undo and Redo in the Node, Rack, and Block editors.

To undo an operation:

- Choose Undo from the Edit menu or from the toolbar.

To redo an operation:

- Choose Redo from the Edit menu or from the toolbar.

Selecting Cells, Rows and Columns

Before you can edit your records in the spreadsheet, you must select the area you want to work with. You can use the first method for all the form/spreadsheet editors. The remaining methods work in the Messages editors only.

All editors

To select a single row:

- Click the row number at the left edge of the spreadsheet.

  The row is highlighted, indicating that it is selected.

Messages only

To select a cell:

- Click the cell, or press TAB or the arrow keys.

  A gray border appears to indicate that the cell is selected.

To select a block of cells:

1. Select a cell in one corner of the block you want to select.
2. Press the left mouse button.
3. While holding down the mouse button, drag the mouse toward the opposite corner of the block until the desired selection is highlighted.
4. Release the mouse button.

   If you are using a keyboard, hold down the SHIFT key and use the arrow keys to highlight the block.

To select more than one row:

1. Click the row number for the first row you want to select.
2. Continue holding the mouse button down and drag the mouse through the row numbers until all the desired rows are highlighted.
3. Release the mouse button.

   The rows are highlighted, indicating that they are selected.

To select a single column:

- Click the column header at the top of the spreadsheet.

  The column is highlighted, indicating that it is selected.

To select more than one column:

1. Click the column header of the first column you want to select.
2. Continue holding the mouse button down and drag the mouse through the column headers until all the desired columns are highlighted.
3. Release the mouse button.

   The columns are highlighted, indicating that they are selected.

To select the entire spreadsheet:

- Choose Select All from the Edit menu, or click on the empty field in the top left corner of the spreadsheet, above the column numbers and to the left of the column headings.

Cutting or Copying and Pasting Selections (Messages Only)

You can cut or copy and paste any entry in the spreadsheet.

Note: You cannot cut, copy, or paste entries in the Node, Rack, and Block editors.

To cut or copy and paste cells, rows or columns:

1. In the spreadsheet, select the area that you want to copy.
2. Choose Cut or Copy from the Edit menu, or or from the toolbar.

   This places the contents in the clipboard, replacing the clipboard's previous contents.
3. Select the area into which you want to paste the contents of the clipboard, or click on a single cell.
4. Choose Paste from the Edit menu or from the toolbar.

   This pastes the selection into the entire area you selected. Or, if you clicked on one cell only, this pastes the selection into the area starting with the cell.

Copying and Pasting into Differently Sized Areas

The area you copy doesn't need to be the same size as the area you paste into.

To copy information into a differently sized area:

1. Select the area you want to copy.
2. Choose Copy from the Edit menu or from the toolbar, or press CTRL+C.
3. Select the area you want to paste into.
4. Choose Paste from the Edit menu or from the toolbar, or press CTRL+V.

If you pasted a small selection into a larger area, information is repeated.

If you pasted a large selection into a smaller area, information is truncated.

Sorting Entries in the Spreadsheet

You can sort all or some of the entries in the spreadsheet numerically or alphabetically. In the Messages editors, if more than one column is selected to be sorted, the Sort operation will proceed beginning with the left-most selected column. Any duplicate items are arranged according to the contents of the next columns.

The Rack editor is sorted numerically by the Rack column. The Block editor is sorted numerically by the File Number column. You cannot use the Sort command in the Node editor; entries are automatically sorted by Node Name when you open the editor.

Important: You can not undo a Sort operation.

To sort all the entries in the spreadsheet:

1. If working in a Messages editor, select the column to sort by clicking on the column heading. You do not need to select a column to sort the Rack and Block editors.
2. Choose Sort or Sort Messages from the Edit menu or from the toolbar.

   For messages, data is sorted numerically if the data in the column is numeric, or alphabetically if the data is text. Entries in the Rack and Block editors are sorted as described above.

Sorting Portions of the Spreadsheet

Messages only

You can sort portions of a spreadsheet in the Messages editors only. The data is sorted according to primary and secondary sort keys.

The left-most selected column is the primary sort key. Each successive selected column is a secondary sort key. That is, the data is first sorted according to the items within the left-most selected column. If there are any duplicate items, they are arranged according to the contents of the next column.

For example, you could sort local messages using the trigger value as the primary sort key. The message text column could be the secondary sort key.

To sort a portion of the spreadsheet:

1. Select the block of fields to sort by clicking on a cell and dragging the mouse diagonally toward the opposite corner of the block.
2. Choose Sort Messages from the Edit menu or from the toolbar.

   Your data is sorted according to the column(s) you selected.

Duplicating Rows

You can repeat the information in a single row by using the duplicate tool. This is useful for creating several rows with similar information.

Note: You can use this command to duplicate rows only, not cells or columns.

To duplicate one or more rows:

1. Select the rows.
2. Choose Duplicate from the Edit menu or from the toolbar, or press CTRL+D.

   The duplicate rows appear immediately below the last row you selected.
3. Since records must be unique, edit each duplicate row.

Filling Portions of the Spreadsheet (Messages Only)

In the Messages editors, there is a quick way to fill portions of the spreadsheet with identical data.

To fill a series of rows with identical data:

1. Select a series of rows.
2. Choose Fill Forward from the Edit menu or from the toolbar, or press CTRL+F.

   The contents of the first row are copied into the successive rows of the selection. Existing data in the successive rows is overwritten. If the first row is blank, all successive rows are blank also.

To fill a series of cells with identical data:

1. Select a group of cells in a column.
2. Choose Fill Forward from the Edit menu or from the toolbar, or press CTRL+F.

   The contents of the first cell or cells is copied down each column into the successive cells. Existing data is overwritten.

   Note: You cannot use Fill Forward in the Node, Rack, and Block editors.

Inserting a Row in the Spreadsheet

You can insert new rows anywhere in the spreadsheet:

1. Select a row in the spreadsheet.
2. Choose Insert from the Edit menu or from the toolbar.

   A blank row is inserted just above the selected row. The existing rows are shifted down.

Deleting Data in the Spreadsheet

You can delete rows, columns, or blocks of data in the Messages editors. You can delete rows of data in the Node, Rack, and Block editors. Unlike the Cut operation, when you delete, the data is not copied to a clipboard.

1. Select the row, column, or block of data you want to delete.
2. Choose Delete from the Edit menu or from the toolbar. Or, press the DELETE key.

   The data is deleted.

   Important: In the Messages editors, use the Undo command to reverse any action since the last update except sorting.

Editing Selections (Messages Only)

In the Messages editors you can edit a group of records or block of cells in one operation. You must have both the form and spreadsheet views open to do this.

1. Select the group of records or block of cells to be edited.

   The first record of the group is displayed in the form section.
2. Edit, in the form section, the field or fields to be changed. Then choose Accept.

   The change is applied to all the selected records, not only to the one displayed in the form.

   Note: All selected cells change to match the corresponding cells in the first selected record. Select only cells you want to change.

Changing Column Width and Row Height

Each time you open a Messages editor, the default column and row settings appear. However, while you are working in the spreadsheet you can customize it by stretching or shrinking both the width of the columns and the height of the rows.

The Node, Rack, and Block editors retain your changes to columns when you close them. For example, hidden columns remain hidden. On the other hand, rows are restored to their default height and position when you reopen the editor.

To change the width of a column:

1. Place the cursor on the line dividing the column headers.

   The cursor changes to a double arrow.
2. Hold the left mouse button down and drag the cursor to stretch or shrink the column width to its desired size.

   You can hide a column completely by shrinking it until it disappears. To retrieve the column, click between the two columns that surround the hidden column in the header, and drag to the right.

To change the height of a row:

1. Place the cursor on the line dividing the row numbers.

   The cursor changes to a double arrow.
2. Hold the left mouse button down and drag the cursor to stretch or shrink the row height to its desired size.

   You can hide a row completely by shrinking it until it disappears. To retrieve the row, click between the two row numbers of the rows above and below the hidden row, and drag downwards. If you forget where the hidden row is, close the editor and reopen it to restore all the rows.

Defining Communications

This chapter describes how to define communications for these network types:

- Remote I/O
- Data Highway Plus
- ControlNet

It also describes how to change your application from one network type to another.

Note: For information about defining communications for a Modbus Application, see the *PanelBuilder 1400e Configuration Software for Windows Modbus User Manual,* Catalog Number 2711E-6.12.

What Are Communications?

The term *communications* refers to the process of exchanging information between the programmable controller and the application on the PanelView terminal. In addition to installing the physical equipment that connects the PanelView terminal to the PLC, you must define communications in PanelBuilder so that the application can send and receive information in the appropriate form when it is running on the PanelView terminal. This chapter describes how to define the communication parameters in PanelBuilder:

- The first section of this chapter describes defining communications for Remote I/O applications.
- The second section describes defining communications for DH+ and ControlNet applications. To define communications for a DH+ or ControlNet application, go to page 5–14.
- The third section of this chapter describes how to change from one communication network type to another. For information about changing from one network type to another, go to page 5–25.

Defining Communications for a Remote I/O Application

To define communications for a Remote I/O application (also called RIO), use these three editors:

- **Communication Setup Editor**—to define the baud rate used to communicate between the terminal and the programmable controller.

  If you are going to transfer the application to the PanelView terminal using Pass-Through via a DH+, ControlNet, or Ethernet network instead of using a serial Upload/Download cable, you also have the option of defining the station address of the terminal's PLC for DH+/RIO, ControlNet/RIO, or RIO/Ethernet Pass-Through, and you can define network file transfer control tags.
- **RIO Racks Editor**—to identify the programmable logic controller the terminal will communicate with, and to identify the racks that are assigned to the terminal. The PanelView terminal supports Full and Listen Only access racks. Racks can be defined as full or partial.
- **RIO Block Transfer Files Editor**—to define block transfer files and the DH+/RIO, ControlNet/RIO, or RIO/Ethernet Pass-Through control byte.

  When assigning addresses in PanelBuilder, you can use either a discrete I/O address or a block transfer file address. Using block transfer files increases the amount of address space available to you for the application. Block transfer files can be defined as PLC input or PLC output files.

The other two editors in the PLC Communications folder, the Scan Classes and Nodes editors, are used exclusively for defining DH+ (Data Highway Plus) and ControlNet communications. See page 5–15.

Communication Setup

The Configure Communication Setup dialog box has two tabs: Terminal Communications and Network File Transfer.

In the Terminal Communications tab, you specify the baud rate used to communicate between the terminal and the PLC, or the terminal network address, depending on the application's network type. The network type is a static field that indicates the network specified for the current application.

To change the application's network type, see "Changing the Network Type," later in this chapter.

The information you define in the Network File Transfer tab is used only for DH+/RIO or ControlNet/RIO Pass-Through file downloads.

If you are going to transfer applications between the development computer and the PanelView terminal using a Serial Upload/Download cable, you do not have to specify anything in the Network File Transfer tab.

To open the Configure Communication Setup dialog box:

- Open the PLC Communications folder and choose Communication Setup, or choose from the toolbar.

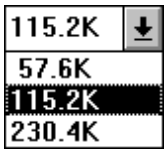
Setting up Terminal Communications

In the Terminal Communications tab you can set the baud rate and the terminal network address.

Setting the Baud Rate

The length of the Remote I/O cable between the terminal and programmable controller dictates the maximum baud rate. For maximum speed, select the highest baud rate that supports your cable length.

For a cable up to	Select this baud rate
2,500 feet	230.4K
5,000 feet	115.2K
10,000 feet	57.6K



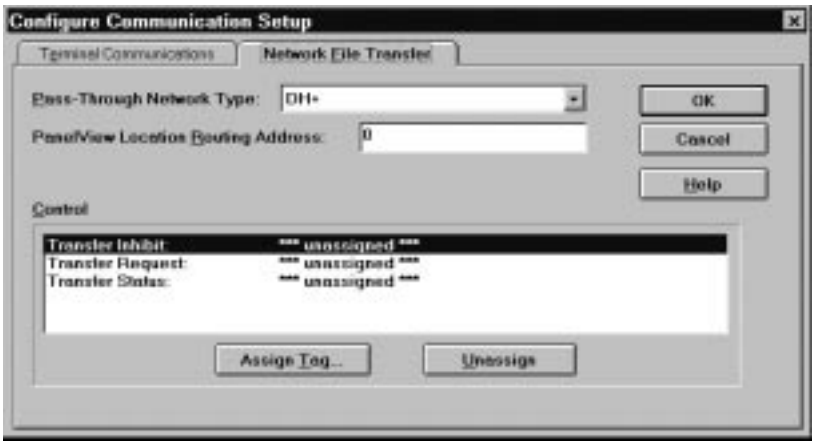
To define the baud rate:

1. In the Configure Communication Setup dialog box choose the Terminal Communications tab.
2. In the Baud Rate drop-down list, choose the desired baud rate. This is the baud rate you noted on the Rack Assignments worksheet.

Note: Not all PLCs support all baud rates. For a list of supported baud rates, consult the User's Manual for your PLC.

Setting Up Network File Transfers

You specify the Pass-Through network type, establish the PanelView Location Routing Address, and define Network Download controls in the Network File Transfer tab.

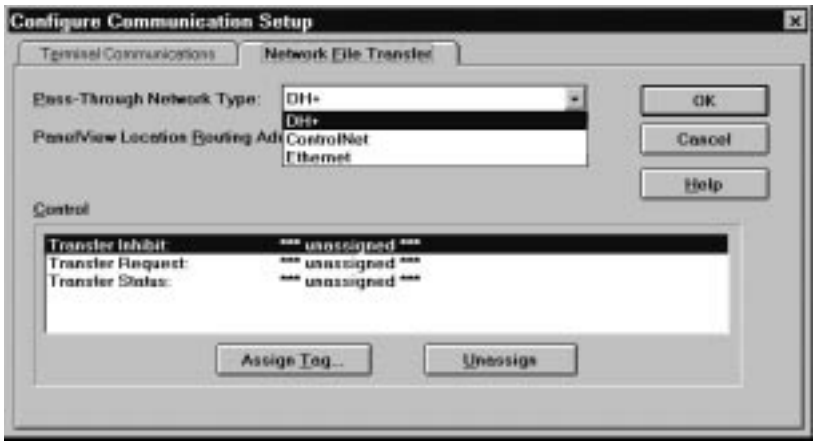


Specifying the Pass-Through Network Type

Remote I/O Pass-Through allows you to download the Remote I/O application file without connecting the serial cable between the PanelView terminal and the development computer. This means you don't have to bring the development computer from terminal to terminal to perform direct downloading. You also don't have to switch the terminal to Configure mode. See Chapter 15, *Transferring Applications*, for more information about performing Remote I/O Pass-Throughs.

To define the Pass-Through network type:

1. Choose the Network File Transfer tab in the Configure Communication Setup dialog box.
2. In the Pass-Through Network Type drop-down list, select the appropriate Pass-Through:
   - ControlNet
   - DH+
   - Ethernet



Specifying the PanelView Location Routing Address

If you are going to download the application to the PanelView terminal using Remote I/O Pass-Through, you should specify the station address of the programmable controller the PanelView terminal is attached to. This address allows PanelBuilder to locate the PanelView terminal.

The address you specify depends on the type of network you are using:

- If the PLC and the development computer are on the same ControlNet network, specify the address of the PLC the terminal is connected to. It must be a decimal number between 1 and 99.
- If the PLC and the development computer are on the same DH+ network, the PanelView Location Routing Address you specify is the PLC's DH+ station address. It will be an octal number between 0 and 77.
- If the PLC and the development computer are on the same Ethernet network, the PanelView Location Routing Address you specify is the PLC's Ethernet station address. It will a number in the form of www.xxx.yyy.zzz where www, xxx, yyy, and zzz are each between 000 and 255.

- If the PLC and the development computer are on different (bridged) DH+ networks, you must specify a remote station address for the programmable controller. Specify both the PLC's station address and the bridge or bridges' addresses.

If you specify a PanelView Location Routing Address, on download of the application this address will be used as the default PanelView Location Routing Address in the Setup Download dialog. This saves you from having to specify the terminal's location each time the application is downloaded. You can change the default if you wish. If you don't specify a PanelView Location Routing Address, you will have to specify one in the Setup Download dialog box.

The following table shows the syntax for a remote station address bridged by a 1785-KA.

Station Address		Valid Entries
ll.rrr.ss		
ll	local bridge address	0 – 77 (octal)
rrr	remote bridge address	0 – 376 (octal)
ss	station address on remote DH+	0 – 77 (octal)
	or	
You can shorten the offlink address by combining the remote bridge and remote station addresses into one three-digit number. The system reads only the first digit of the remote bridge address, and adds the two-digit remote station address to produce a shorter address that still points to the same node.		
ll.rss		
ll	local bridge address	0 – 77 (octal)
rss	combined remote bridge address and remote station address	0 – 376 (octal)

The following table shows the syntax for a remote station connected by a Pyramid Integrator.

Station Address		Valid Entries
ll:rrrrr.ss		
ll	local bridge address	0 – 77 (octal)
rrrrr	PI Link number	0 – 65535 (decimal)
ss	station address on remote DH+	0 – 77 (octal)

For examples of valid remote station addressing, see the section "DH+ Addressing" in Chapter 2, *Planning Applications*. Though the examples in Chapter 2 show remote station addresses for a PLC, the method is the same for the PanelView Location Routing Address.

To define the PanelView Location Routing Address:

1. In the Configure Communication Setup dialog box, choose the Network File Transfer tab.
2. Type the PLC's station address in the PanelView Location Routing Address field**.**

Defining the Network File Transfer Controls

If you plan to download an application via DH+ or ControlNet networks using the Pass-Through Download feature, you can define the network file transfer controls to monitor file download processes.

Note: Although it's optional, we strongly recommend that you define and use these controls. You must assign all three controls for the mechanism to work properly.

There are three network download controls. You can assign tags only to these controls:

- **Transfer Inhibit**—The terminal reads this control at the start of a download. When this control is set to non-zero, the terminal won't permit a download.

  You can use a digital tag for the Transfer Inhibit control since the terminal only looks for 0 and non-zero. Typically, the control will be set and cleared by the programmable controller.
- **Transfer Request**—This control is set to 1 by the terminal when it receives a download request. It is reset to 0 by the terminal when the download starts. This control should use a digital tag that can be written to by the terminal.

  You can program the PLC to clear the Transfer Inhibit control when this control is set to 1. This will allow subsequent file downloads to proceed.
- **Transfer Status**—This control is set to 1 by the terminal when it receives the first download block transfer. It remains set throughout the transfer and is reset to 0 when the download is completed, when the terminal's mode is switched or when the power is cycled. The Transfer Request control is cleared when the Transfer Status control is set. This control should use a digital tag that can be written to by the terminal.

For more information on using Remote I/O Pass-Through to download applications, see Chapter 15, *Transferring Applications*.

To define network file transfer controls:

1. In the Network File Transfer tab of the Configure Communication Setup dialog box, select one of the controls and click Assign Tag to choose a tag. You can assign a digital or analog tag, although a digital tag is recommended.

   The Tag Browser dialog box appears.

2. Select a tag from the Tags list, or choose New Tag to create a new tag. Refer to Chapter 6, *Defining Tags*, for more detailed information about the Tag Editor dialog box.

3. Repeat Steps 1 and 2 for the other two controls.

Assigning Device Type and Racks for the PanelView Terminal

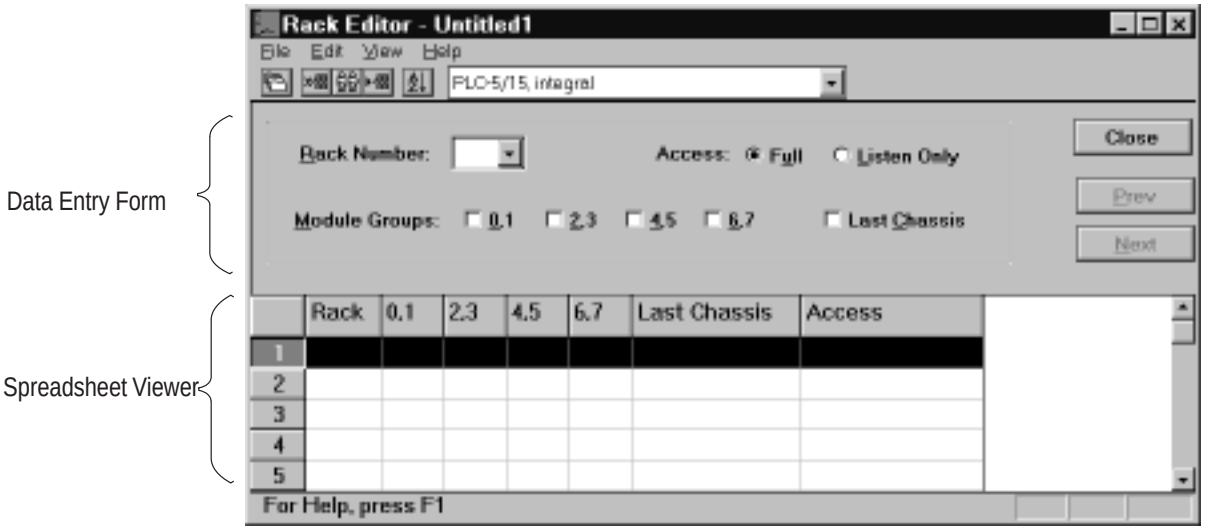
Before you can define tags for objects, windows, or PLC I/O Control options, you need to define the device type and racks that these tags will address. This is done in the RIO Racks editor, using the Rack Assignments worksheet you prepared in Chapter 2, *Planning Applications*.

Important: When you change the rack number of a rack assignment already used by a tag, Block Transfer File, or Pass-Through Control Byte, all references to the rack are updated to reflect the new number.

To open the Rack Editor dialog box:

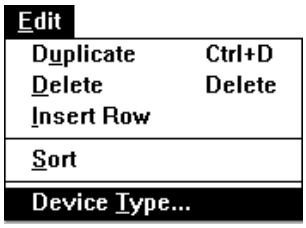
- Open the PLC Communications folder and choose RIO Racks, or choose from the toolbar.

The Rack editor has two parts: the data entry form on top where you define the rack, and the spreadsheet viewer on the bottom where you see all the rack assignments and select one for editing.



Specifying the Device Type

Before you can assign racks, you must first specify the type of programmable controller the terminal will be connected to. The default is the PLC-5/15.



To specify the PLC type:

- Select a device type from the drop-down list on the Rack Editor tool bar, or choose Device Type from the Edit menu and select the type to use.

Addresses of tags used in an application must conform to the address syntax for the selected device. The SLC address syntax differs from that of the PLC devices. When the device is switched from a PLC to SLC device or vice versa, the application tag addresses will be updated automatically when the application is validated.

Tip: Validate your application right after a SLC to PLC or PLC to SLC device type change to update all your tag addresses so your reports and the Tag Usage Viewer will display the correct information.

Assigning Racks

You can define the PanelView terminal to function as one or more partial or full racks on the programmable controller. You can also configure the terminal to monitor the communication between the programmable controllers and other Remote I/O adaptors.

You can have up to 64 rack assignments. Each will give you both input and output address space. To assign racks, you must define the following for each rack assignment:

- **Rack Number**—Any rack supported by your programmable controller can be used in a rack assignment. You can make several rack assignments with the same rack number, until you have assigned all module groups for that rack.
- **Access**—Access can be Full or Listen Only.

  Specify Full access for racks whose I/O space is in the terminal. The input section of a Full access rack can be written to by the terminal.

  Listen Only access means that the PanelView terminal can read from the rack's I/O space, but can't write to it. Listen Only is useful when one PanelView application needs to monitor the I/O activity on racks used by other devices or configured with Full access for another PanelView terminal.

  Note: You can't configure Listen Only for Rack 0 of PLC-2 and PLC-5 programmable controllers.

- **Module Groups**—You can select any or all module groups (0, 2, 4, or 6). Each of these module groups occupies a 1/4 rack. Follow these guidelines when assigning module groups:
  - Assign each module group only once for any given rack that has multiple assignments.
  - Ensure module group assignments are contiguous for each rack assignment. For example, you can assign modules 0, 2, and 4, but not modules 0, 2, and 6.
- **Last Chassis**—Enabling Last Chassis signals to the programmable controller that it does not need to scan further module groups for the racks past those defined in the rack assignment. If you are using partial rack assignments, you can enable Last Chassis for any one of module groups 0,1 or 2,3 or 4,5. Module group 6,7, if used in a rack assignment, will always have last chassis enabled automatically.

To assign racks:

1. In the Rack Number field, select the rack number you want to use.
2. Select the contiguous module groups you want to assign for that rack number. An X appears in the check box beside each module group you select.

   Note: For any rack number there can be only one rack assignment with Last Chassis enabled. When you define multiple assignments for a rack, any Last Chassis enabled for those with lower module groups will be disabled when rack assignments with higher module group numbers are defined.
3. When you have finished, choose Accept to save the rack assignment.
4. If desired, assign further racks and related module groups.

Setting Up Block Transfer Files

A tag with a block transfer file address can be used anywhere in place of a tag with a discrete I/O address.

Discrete addressing provides better performance and requires less PLC programming. However, if you don’t have enough discrete address space, or need more contiguous address space, use block transfer files.

A block transfer file can be either a read or a write file, with Full or Listen Only access, depending on the access of the assigned control byte. A byte is used in the output space of the rack, regardless of file type. However, a read and a write file can share the same byte in the output table.

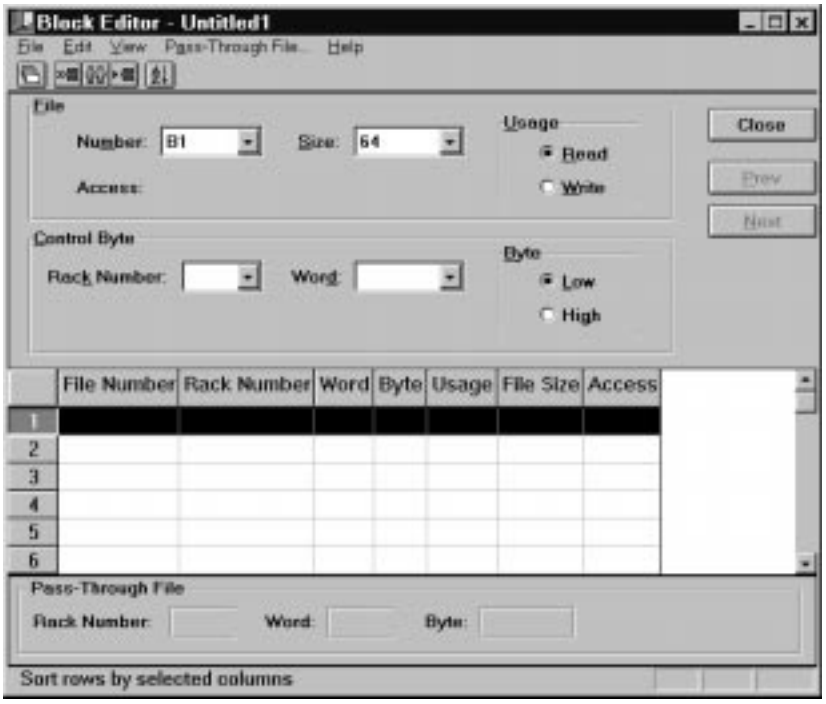
Important: Some scanners (the 1771-SN and 1747-SN Series A modules, for example) don't support block transfers to a Remote I/O device. Block transfer file assignments apply only if you're using another remote 1771 I/O or 1747-SN Series B scanner device. See your scanner's User Manual to determine whether it supports the use of block transfers.

Each block transfer file must be assigned a control byte. This control byte occupies space on an already defined rack. In other words, before you can assign block transfer files, you must define the rack assignments. See the previous section for more information. Use the Block Transfer File Assignments worksheet you created in Chapter 2, *Planning Applications.*

To open the RIO Block Transfer Files editor:

- Open the PLC Communications folder and choose RIO Block Transfer Files, or choose from the toolbar.

The Block Editor dialog box appears.



The Block editor has two parts: the data entry form on top where you define the block transfer file and the spreadsheet viewer on the bottom where you can see all the block transfer files and select one for editing. See Chapter 4, *Working with Editors*, for details on working in the form and spreadsheet sections of this editor.

Creating a Block Transfer File

You can have up to 64 input or output block transfer files, numbered B1 to B64. In the Block editor you can also define the Remote I/O Pass-Through control byte for application file transfer. Fill in the information from the worksheet in the steps that follow.

To define a block transfer file:

1. In the File area of the editor, select the file number and size for the block transfer file.
   - **File Number**—Specify the block transfer file number (a number between B1 and B64).
   - **File Size**—This refers to the number of words in the block transfer file. The default is 64. You can assign any number up to 64.
   - **Access**—The access (Full or Listen Only) is determined by the access of the file's control byte. This field is updated automatically when you define the control byte's address.

   ▶ **Note:** The terminal can only write to a Full Access Input block transfer file.

2. For each block transfer file, you must define a control byte. This byte uses space in the discrete output table only. In the Control Byte area of the editor, choose the rack number, word, and byte for the control byte, and specify whether it will be used for a read or a write block transfer.

   Important: You cannot enter the same control byte for tags or for the Pass-Through File control byte.

   - **Rack Number**—The rack numbers available for each file depend on the racks you assigned earlier. Specify which rack will contain the control byte.
   - **Word**—Specify which word (between 0 and 7) in the specified rack will contain the control byte.
   - **Byte**—Specify which byte of the word to use for the control byte, the high byte or low byte.
   - **Usage**—Specify whether the block transfer byte is a block transfer read or a block transfer write.
3. Choose Accept.
4. Define any other files you will need.

Defining the Pass-Through File Control Byte

Before you can transfer an application file over the DH+, ControlNet, or Ethernet network using the Remote I/O Pass-Through, you must define the Pass-Through File Control Byte. You configure this byte in the RIO Block Transfer Files editor.

Note: If the PLC specified as the device in the RIO Racks editor does not support pass-through, you will not be able to define a Pass-Through File control byte. If this happens, go into the RIO Racks editor and select another device type.

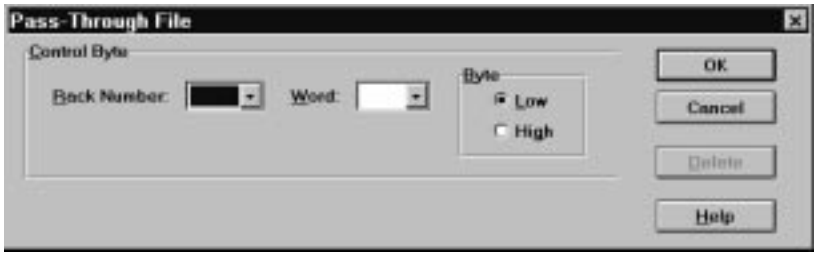
In addition to the 64 available block transfer files, there is a dedicated block transfer file, called Pass-Through. The file size for this block transfer is fixed at 64 words, and can't be changed.

To define the Pass-Through File Control Byte:

Fill in the information from the Block Transfer File Assignments worksheet.

1. Choose Pass-Through File from the Block editor's menu.

   The Pass-Through File dialog box appears.



2. Choose the rack number, word, and byte for the Pass-Through File's control byte. The rack must have full access.

   A control byte is allocated in the output file. You can't use this byte for a tag address or for the control byte of any other block transfer files.

3. Choose OK.

Changing Rack or Block Transfer File Assignments

As you develop the application, you may find that you need to change the rack or block transfer file assignments. When you change a rack assignment, any assigned tags, Block Transfer Files, or Pass-Through File control bytes are updated. You can also add new racks, module groups, and/or block transfer files as required.

To change an existing rack or block transfer file assignment:

1. Open the RIO Racks editor or RIO Block Transfer Files editor.
2. Click on the row in the spreadsheet to be changed, or choose Next in the form section until the desired row is highlighted in the spreadsheet.
3. To change the assignment, select or deselect the appropriate fields in the form section.
4. Choose Accept in the form section to confirm the new rack or block transfer file assignment.

   A message informs you that all tags, block transfer files, and Pass-Through control bytes that reference the original rack will be changed to the new rack number.
5. When you're finished changing rack or block transfer file assignments, choose Close.

You can delete racks or module groups if they haven't been used in a tag or block transfer file definition.

If you try to delete a rack that has tags or block transfer files assigned to it, you will see an error message. You must reassign the tag addresses and/or block transfer files to another rack before you can delete the rack to which they're assigned.

To delete an existing rack or block transfer file assignment:

1. Open the RIO Racks editor or RIO Block Transfer Files editor.
2. Click on the row in the spreadsheet to be deleted, or choose Next in the Form section until the desired row is highlighted in the spreadsheet.
3. Choose Delete from the Edit menu.
4. When you're finished, choose Close.

To delete a module group from a rack:

- ▶ Click the check box beside the module group to clear the X from it. Then click Accept to save the change to the rack assignment.

Defining Communications for a Data Highway Plus or ControlNet Application

To define communications for a DH+ or ControlNet application, use these three editors:

- **Communication Setup Editor**—to specify the address of the PanelView terminal on the DH+ or ControlNet network.

  If you are going to transfer the application to the PanelView terminal over the DH+ or ControlNet network instead of using a serial Upload/Download cable, you can also define the terminal's complete DH+ or ControlNet routing address and network download control tags.
- **Node Editor**—to identify the type and address of the programmable logic controllers the application will communicate with.
- **Scan Class Editor**—to define the speed at which tags used in an application will be scanned for new values.

The other two editors in the PLC Communications folder, the RIO Racks and RIO Block Transfer Files editors, are used exclusively for defining Remote I/O communications.

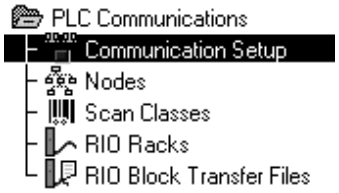
Communication Setup

The Configure Communication Setup dialog box has two tabs: Terminal Communications and Network File Transfer.

In the Terminal Communications tab, you specify the address of the PanelView terminal on the DH+ or ControlNet network.

If you are going to transfer the application between the development computer and the PanelView terminal using a serial Upload/Download cable, you do not have to specify anything in the Network File Transfer tab.

If you are going to transfer the application via the DH+ or ControlNet network, you can specify the complete address for the PanelView terminal. This address is used as the default terminal location when the application is downloaded over the network. You should also assign network download control tags to facilitate the transfer of the application file. Do this in the Network File Transfer tab.



To open the Configure Communication Setup dialog box:

- Open the PLC Communications folder and choose Communication Setup, or choose from the toolbar.

Setting Up Terminal Communications

In the Terminal Communications tab you can set the PanelView terminal network address. For DH+ applications, PanelBuilder 1400e only supports a baud rate of 57.6K.

Note: If you want to communicate on DH+ at 230K baud, you must configure the baud rate from the Terminal Communications screen on the PanelView terminal, after you have downloaded the application to the terminal. See the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual* for more information about configuring the Terminal Communications screen.

Specifying the Network Type

The Network Type input field displays the network type selected for the current application when you created it. This refers to the network through which the terminal and the programmable controllers will communicate. You can change this by selecting another type from the drop-down list. See "Changing the Network Type" on page 5–25 for details.

Setting the Terminal Network Address

The Terminal Network Address lets you specify the PanelView terminal's DH+ or ControlNet address when it is running this application.

By default, the Terminal Network Address is assigned to the PanelView Location Routing Address field in the Network File Transfer tab.

ControlNet

For ControlNet applications, because no ControlNet bridge devices are available, the PanelView Location Routing Address can only be a local network address.

DH+

For DH+ applications, if you change the Terminal Network Address, PanelBuilder will update the PanelView Location Routing Address to match, and vice versa. A message will be displayed informing you of the change. Any bridged DH+ addresses in the PanelView Terminal Address will not be affected.

When you set up a DH+ or ControlNet Direct download, the Network File Transfer's PanelView Location Routing Address is used. This ensures that the download does not go to the wrong station by mistake.

To set the Terminal Network Address:

- In the Terminal Network Address field, type the PanelView terminal's address on the network. For DH+ networks, valid entries are 0 to 77 (octal). For ControlNet networks, valid entries are 1 to 99 (decimal).

  Important: For versions 1.0 and 1.25 of ControlNet, the PanelView Terminal Network Address should not be set to 1.

Setting Up Network File Transfers

You establish the PanelView Location Routing Address and define network download control tags in the Network File Transfer tab.

Specifying the PanelView Location Routing Address

ControlNet

For ControlNet applications, because no ControlNet bridge devices are available, the PanelView Location Routing Address can only be a local network address.

DH+

If you are going to transfer the application to the PanelView terminal over DH+ directly, rather than using a serial Upload/Download cable, you must specify the PanelView Location Routing Address so that PanelBuilder can locate the PanelView terminal. By default, the PanelView Location Routing Address defined in the Network File Transfer tab is used.

If the PanelView terminal and the development computer are on the same DH+ network, the PanelView Location Routing Address will be the terminal's DH+ station number. It will be an octal number between 0 and 77.

If they are on different (but connected) DH+ networks, you must also specify a local and remote bridge address.

The following table shows the syntax for a remote station address bridged by a 1785-KA.

Station Address		Valid Entries
ll.rrr.ss		
ll	local bridge address	0 – 77 (octal)
rrr	remote bridge address	0 – 376 (octal)
ss	station address on remote DH+	0 – 77 (octal)
	or	
You can shorten the offlink address by combining the remote bridge and remote station addresses into one three-digit number. The system reads only the first digit of the remote bridge address, and adds the two-digit remote station address to produce a shorter address that still points to the same node.		
ll.rss		
ll	local bridge address	0 – 77 (octal)
rss	combined remote bridge address and remote station address	0 – 376 (octal)

The following table shows the syntax for a remote station connected by a Pyramid Integrator.

Station Address		Valid Entries
ll:rrrrr.ss		
ll	local bridge address	0 – 77 (octal)
rrrrr	PI Link number	0 – 65535 (decimal)
ss	station address on remote DH+	0 – 77 (octal)

For more examples of valid remote station addressing, see the section "DH+ Addressing" in Chapter 2, *Planning Applications*. While these examples show remote station addresses for a programmable controller, the method is the same for the PanelView Location Routing Address.

To define the PanelView Location Routing Address:

1. In the Configure Communication Setup dialog box choose the Network File Transfer tab.
2. In the PanelView Location Routing Address field, type the terminal's station number. If the PanelView terminal and development computer are on different, bridged DH+ networks, you must specify local and remote bridge addresses, in addition to the station number.

Defining the Network File Transfer Controls

If you plan to download your application via DH+ or ControlNet networks using a Direct download or a Remote I/O download, you should define the network file transfer controls. They allow you to control and monitor the download process, and to communicate the status of the PanelView terminal to other devices.

Important: You must assign all three controls for the mechanism to work properly.

There are three network download controls. You can assign tags only to these controls:

- **Transfer Inhibit**—When this control is set to non-zero, the terminal won't permit a download.

  We recommend you assign a digital tag type. The tag's PLC address will be read by the terminal. Typically, it will be set and cleared by the PLC.

- **Transfer Request**—This control is set to 1 by the terminal when it receives a download request. The control is reset to 0 when the download starts. If a file download is prevented by the Transfer Inhibit control, this control will be set to 1. You can program the PLC to clear the Transfer Inhibit control when this control is set to 1. This will allow subsequent file transfers to proceed.

  We recommend you assign a digital tag type. The terminal will write to the tag's PLC address.

- **Transfer Status**—This control is set to 1 by the terminal when it receives the first download block transfer. It remains set throughout the transfer and is cleared (reset to 0) when the download is completed successfully. The Transfer Request control is cleared when the Transfer Status control is set.

  We recommend you assign a digital tag type. The tag's PLC address will be read and written to by the terminal.

For more information on downloading, see Chapter 15, *Transferring Applications*.

To define network file transfer controls:

1. In the Network File Transfer tab, select one of the controls and click Assign Tag to choose a tag. You can assign a digital or analog tag, although a digital tag is recommended.

   The Tag Browser dialog box appears.

   Tip: You can also double-click the selected address to bring up the Tag Browser.

2. Select a tag from the list, or choose New Tag to create a new tag. Refer to Chapter 6, *Defining Tags*, for more detailed information about the Tag Editor dialog box.

3. Repeat for the other two controls.

Configuring Nodes

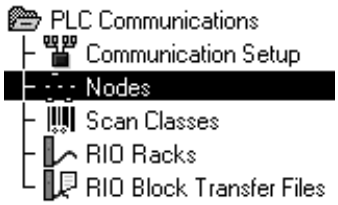
A node is a programmable logic controller connected to the DH+ or ControlNet network. When you configure nodes, you specify a name, type, and station address for each programmable controller with which the application will communicate. A device at a DH+ or ControlNet station is referred to throughout a PanelView application by its node name.

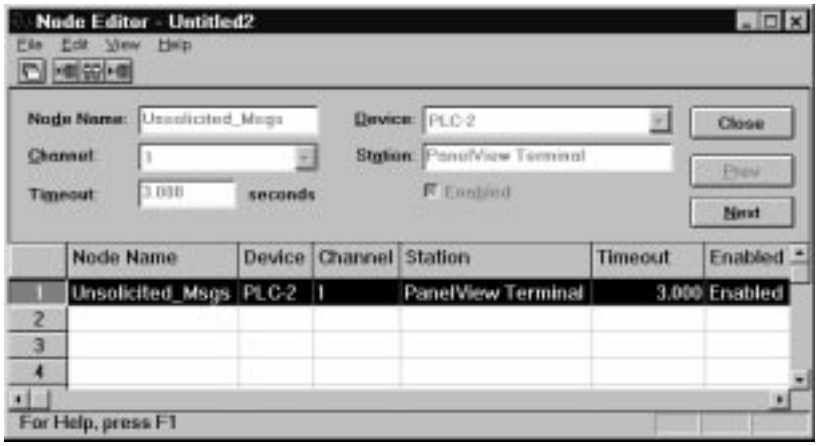
You should already have determined the station address for each node when you completed the Communications Worksheet. If not, see Chapter 2, *Planning Applications*, for details on determining station addresses.

Note: A node definition can be changed at any time during application development. At application or screen validation, or download, the nodes used by the tags in the application must be defined. If you change the node type, the tags that use this node may have invalid addresses. If you change the node name, you can automatically update your tags at the same time.

To open the Node editor:

- Open the PLC Communications folder and choose Nodes, or choose from the toolbar. The Node editor opens.





Data Entry Form

Spreadsheet Viewer

The Node editor has two parts: the data entry form on top where you define the node, and the spreadsheet viewer on the bottom where you see all the nodes and select one for editing.

To configure the nodes:

1. Fill in the following information for each node. Use the information you recorded in the Communications Worksheet.

   Node name
   Type a name of your choice up to 40 characters long. This name represents a programmable controller on the network. The only valid characters for a node name are a to z, A to Z, 0 to 9, - (dash), and _ (underscore). The name is not case-sensitive; your characters will be displayed as you typed them. Spaces are not permitted.

   ▶ **Note:** The node name "Unsolicited_Msgs" is reserved for system use in DH+ applications and cannot be assigned to a node defined by the user. It allows the terminal to support unsolicited messages to and from the PLC. You cannot use a node with this name in ControlNet applications.

   ▶ **Tip:** Use a name that will help you to identify your programmable controllers when you are assigning tags later in your application. For example, you could name your programmable controllers according to their locations or according to the machine or part of the process they will be controlling or monitoring.

   Type
   Select the type of programmable controller you are using from the drop-down list. The type of programmable controller determines the addressing syntax and range for all the tags with addresses on this node.

   ControlNet For ControlNet applications, you can only select PLC-5 (Enhanced).

   Channel
   The Channel number is 1 by default. The application network type determines this channel. You cannot change this value. This field maintains compatibility with RSView™ and provides future communication option expansion.

   Station
   Type the address of the programmable controller.

   DH+ You can use local and remote stations on bridged DH+ networks. For station addressing syntax, see the section "DH+ Addressing" in Chapter 2, *Planning Applications*.

   ControlNet For ControlNet applications, values are between 1 and 99 (decimal). Only local addresses are supported.

Timeout
Type the number of seconds (0.100 to 6553.500) you want the terminal to wait before reporting a communication error. The default timeout period is three seconds.

Enabled
Normally nodes are enabled, allowing data collection. However, while you are debugging your application or troubleshooting your network, you can disable a node to prevent communication faults or invalid data. Once the application file is downloaded and running, any values normally sent to a disabled node will be recorded in the terminal, but the actual PLC value will remain unchanged.

To disable nodes, clear the Enabled check box.

Important: Make sure the nodes are enabled before you run an application in an online environment.

2. When you've finished configuring nodes, choose Close.

Unsolicited Messages (DH+)

For DH+ applications, the PanelView terminal supports Unsolicited Messages using the PLC-2 format, unprotected read and write requests. Unsolicited messages allow the terminal to talk to more than one programmable controller using only a single tag address. The terminal will maintain a PLC-2 format data table of 512 words. Data is retained across power-cycles and mode switches.

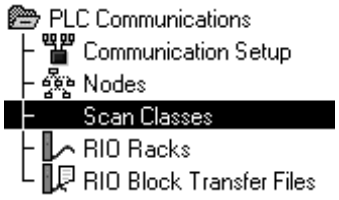
At runtime, a PanelView application can interact with programmable controllers using Unsolicited Messages by defining tags on the Unsolicited_Message node, with addresses in the 512-word data table. When the application runs, new values written to the tag are stored in the terminal's data table where they can be read by other programmable controllers. In addition, the programmable controller can write values to the terminal's data table and PanelView will scan the tag addresses to read the new values.

Tags defined on the Unsolicited_Message node must use standard PLC-2 addressing format. See Appendix C, *Addressing Syntax*, for addressing format information.

Configuring Scan Classes

Any tag that has Device as its data source must be assigned to a scan class. At runtime, the PanelView terminal periodically updates its tag information by scanning the programmable controller addresses. You determine how often a tag value is updated by assigning the tag to a scan class that you have configured in the Scan Classes editor. Make sure that tags with rapidly changing PLC values are assigned to a fast scan rate, while those that don't change often are scanned less frequently. This helps optimize traffic flow on the DH+ or ControlNet network.

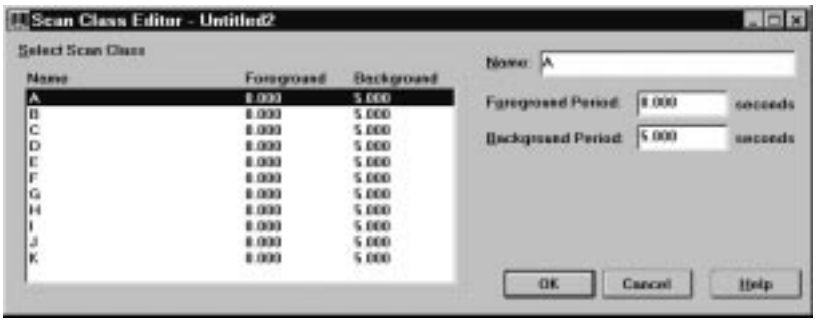
You should already have determined the rates for the scan classes when you completed the Communications Worksheet. If not, see Chapter 2, *Planning Applications*, for tips on assigning foreground and background scan rates.



To open the Scan Class editor dialog box:

- Open the PLC Communications folder and choose Scan Classes, or choose from the toolbar.

  The Scan Class editor opens.



To configure scan classes:

1. Fill in the following information for each scan class. Use the information you recorded in the Communications Worksheet.

   Name—A Through K
   You can keep the letter for the name of the scan class or you can type a name up to 20 characters long. The name can have upper and lower case letters, numbers, dashes, and underscores. Spaces are not permitted.

   If you type a meaningful name, it will be easier to keep track of what the scan classes represent. For example, you might name one scan class TimeCritical, another SlowChanging, and so on.

Foreground and Background Period
The period specifies, in seconds, how often the programmable controller address will be scanned. A foreground period applies to tags used on the current screen, to the alarm and information window tags, foreground and background trends, and to the PLC-controlled PLC I/O Control tags. The background scan period applies to tags used in all other screens in the system. The valid range of values is 0 to 86,400 seconds (24 hours), with a 0.1 second resolution.

To specify a period, type a number. You can use fractions of a second. For example, 0.6 means scan every six-tenths of a second. Keep the background rate slow so as not to slow down network traffic.

Tip: For more information about optimizing ControlNet or DH+ highway traffic, refer to your network user documentation. Also see "Guidelines for Configuring Scan Classes" in Chapter 2, *Planning Applications.*

Note: If a tag is being used by both a foreground and background component, the tag will be scanned at the foreground rate.

2. To save the information you've configured, choose OK.

Note: If you change the name of a scan class used by tags, you have the option of automatically updating the tags with the new name.

Changing the Network Type

When you create an application, you specify the type of network it is to run on, and set up the communication parameters accordingly. You can change the network type of an existing application to run it on another network. You do this using the Change Network Type dialog box.

However, when you change the network type, all communication specifications, including node definitions and scan classes, will be lost. You will have to re-configure communications for your application.

This section describes the following network switches:

- converting from Remote I/O to DH+
- converting from Remote I/O to ControlNet
- converting from DH+ to Remote I/O
- converting from DH+ to ControlNet
- converting from ControlNet to Remote I/O
- converting from ControlNet to DH+

Note: You can either convert the application only, or the application and its tags. If you convert the tags, edit their addresses to ensure they are valid for the new network type.

Note: For information about converting applications to and from Modbus, see the *PanelBuilder 1400e Configuration Software for Windows Modbus User Manual,* Catalog Number 2711E-6.12.

Converting from Remote I/O to DH+

When you change an application from Remote I/O to DH+, you will have to define the necessary DH+ nodes and scan classes and change any tag addresses that are not valid for DH+ device tags. For more information, see "Defining Communications for a Data Highway Plus or ControlNet Application," earlier in this chapter.

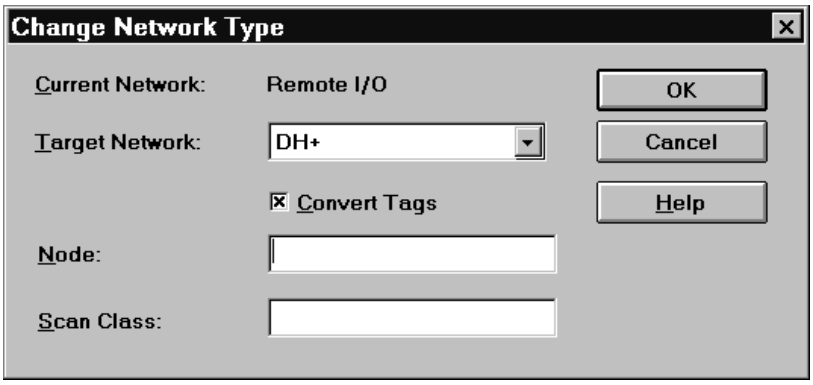
To change the network type from Remote I/O to DH+:

1. Choose Change Network Type from the Tools menu.
2. In the Target Network field, choose DH+.
3. To convert the tags in the application, select the Convert Tags check box. If you don't want to convert tags in the application, clear the check from the Convert Tags check box.
   - If you don't convert the tags to DH+ format, the tags' data source remains Remote I/O. For any tags to be used in the DH+ application, you will have to change the data source to Device and give them valid DH+ addresses.
   - If you do convert the tags to DH+ format, all the tags in the database that are used by your application will be converted to Device tags. Analog Tags using the binary data type will be switched to unsigned integer because binary is not supported for DH+ applications.

Note: If your application's Target Firmware version is Version 1 or Version 1.03, block tags used in the application are reported as errors when you validate the application. Block tags are supported in DH+ and ControlNet applications that use Target Firmware Version 2 or later.

Note: If you convert shared tags, this may make them invalid for the other applications that are using them. Also, unused tags are not converted and remain as Remote I/O tags.

Note: If you convert an application from Remote I/O to DH+ or ControlNet, or vice-versa, the tag addresses may be invalid for the network to which you are converting. Change the tag addresses to match the network type.



4. Specify a node name and scan class.

You can specify a node name and scan class when you do the conversion, or you can leave them blank. If you enter a node name and scan class, they will be used in all the tags that are converted. In addition, the node and scan class will be created for you with the default settings. You can change the settings to suit your application later.

If you leave the Node Name and Scan Class fields blank, the converted tags will have a blank node and scan class name. You will have to create the nodes and scan classes that your tags will use, then update each tag with the node and scan class names that you have created. For details, see "Defining Communications for a Data Highway Plus or ControlNet Application," earlier in this chapter.

5. Choose OK.

The following message appears:

Configure Communication Setup
Rack and Block Transfer File definitions will be lost after the network type is converted to DH+. Continue?
Yes
No
Help

6. Choose Yes to confirm that you want to continue.

Converting from Remote I/O to ControlNet

To change the network type from Remote I/O to ControlNet:

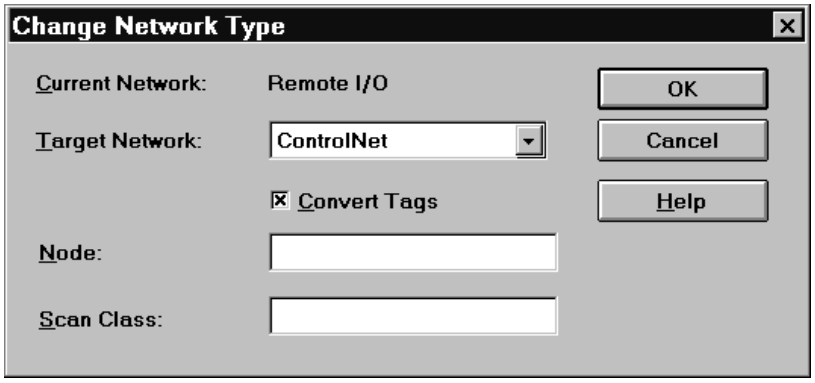
Important: When you change an application from Remote I/O to ControlNet, you will have to define the necessary ControlNet nodes and scan classes and change any tag addresses that are not valid for ControlNet device tags.

1. Choose Change Network Type from the Tools menu.
2. In the Target Network field, choose ControlNet.
3. To convert the tags in the application, select the Convert Tags check box. If you don't want to convert tags in the application, clear the check from the Convert Tags check box.
   - If you don't convert the tags to ControlNet format, the tags' data source remains Remote I/O. For any tags to be used in the ControlNet application, you will have to change the data source to Device and give them valid ControlNet addresses.
   - If you do convert the tags to ControlNet format, all the tags in the database that are used by your application will be converted to Device tags. Analog tags using the binary data type will be switched to unsigned integer because binary is not supported for ControlNet applications.

▶ **Note:** If your application's Target Firmware version is Version 1 or Version 1.03, block tags are reported as errors when you validate the application. Block tags used in the application are supported in ControlNet applications that use Target Firmware Version 2 or later.

▶ **Note:** If you convert shared tags, this may make them invalid for the other applications that are using them. Unused tags are not converted and remain as Remote I/O tags.

▶ **Note:** If you convert an application from Remote I/O to DH+ or ControlNet, or vice-versa, the tag addresses may be invalid for the network to which you are converting. Change the tag addresses to match the network type.



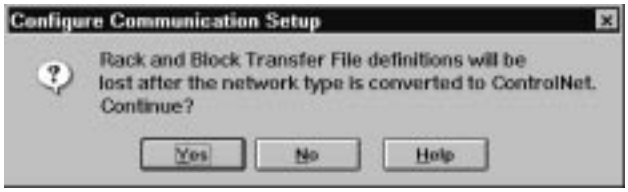
4. Specify a node name and scan class if required.

   You can specify a node name and scan class when you do the conversion, or you can leave them blank. If you enter a node name and scan class, they will be used in all the tags that are converted. In addition, the node and scan class will be created for you with the default settings. You can change the settings to suit your application later.

   If you leave the Node Name and Scan Class fields blank, the converted tags will have a blank node and scan class name. You will have to create the nodes and scan classes that your tags will use, then update each tag with the node and scan class names that you have created. For details, see "Defining Communications for a Data Highway Plus or ControlNet Application," earlier in this chapter.

5. Choose OK.

   The following message will appear:


Configure Communication Setup
Rack and Block Transfer File definitions will be lost after the network type is converted to ControlNet. Continue?
Yes
No
Help

6. Choose Yes to confirm that you want to continue.

7. After converting the application to ControlNet, you need to change the Terminal Network Address on the Terminal Communications tab of the Configure Communication Setup dialog.

Since Remote I/O does not make use of a terminal network address, this field contains the default terminal address, which is 2. For a ControlNet network, the terminal network address must be between 1 and 99 (decimal). Ensure the value is correct for your application.

Important: For versions 1.0 and 1.25 of ControlNet, the PanelView Terminal Network Address should not be set to 1.

In the Network File Transfer tab, the PanelView Location Routing Address will be set to the Terminal Network Address.

For details on configuring an application for ControlNet networks, see "Defining Communications for a Data Highway Plus or ControlNet Application," earlier in this chapter.

Converting from DH+ to Remote I/O

If you change an application from DH+ to Remote I/O, you will have to define the necessary Racks, specify the Block Transfer File and Pass-Through control bytes, if needed, and change any tag addresses that are not valid for Remote I/O.

To change the network type from DH+ to Remote I/O:

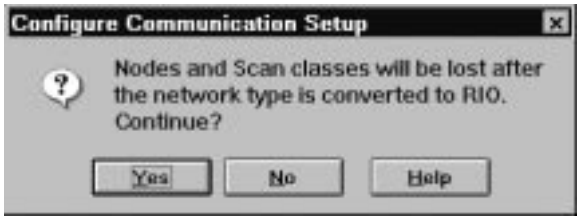
1. Choose Change Network Type from the Tools menu.
2. In the Target Network field, choose Remote I/O.
3. To convert the tags in the application, select the Convert Tags check box. If you don't want to convert tags in the application, clear the check from the Convert Tags check box.
   - If you don't convert the tags to Remote I/O format, the tags' data source remains Device and the address is unchanged. You will have to change the data source to Remote I/O manually for any of these tags that you want to use, and give them each a Remote I/O address.
   - If you convert the tags to Remote I/O, the data source for all the tags is changed to Remote I/O, but you will still have to give them valid Remote I/O addresses manually.

Note: Converting tags that are shared may make them invalid for the other applications that are using them. Unused tags that are not converted will stay as DH+ tags.

Note: If you convert an application from Remote I/O to DH+ or ControlNet, or vice-versa, the tag addresses may be invalid for the network to which you are converting. Change the tag addresses to match the network type.

4. Choose OK.

   The following message appears:


Configure Communication Setup
Nodes and Scan classes will be lost after the network type is converted to RIO. Continue?
Yes
No
Help

5. Choose Yes to confirm that you want to continue.
6. Once you have changed the network type you will have to configure the programmable logic controller type, rack assignments, block transfer files, and Pass-Through File control byte, if needed, in the Rack and Block Transfer File editors.

For details on configuring an application for Remote I/O, see "Defining Communications for a Remote I/O Application," earlier in this chapter.

Converting from DH+ to ControlNet

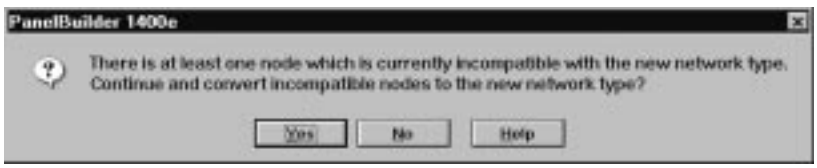
To change the network type from DH+ to ControlNet:

1. Choose Change Network Type from the Tools menu.
2. In the Target Network field, choose ControlNet.
3. Choose OK.

   The following message will appear:

4. Choose Yes to change your DH+ application to ControlNet. Choose No if you do not wish to continue with the conversion.

   Note: If a DH+ node is defined as station 0, or station 64 through 77 (inclusive), it will be invalid for a ControlNet application. The following message will appear, prompting you to convert the station number.


PanelBuilder 1400e
There is at least one node which is currently incompatible with the new network type. Continue and convert incompatible nodes to the new network type?
Yes
No
Help

5. Select Yes or No.

   If you select Yes, any node with an invalid station number for the ControlNet network will be changed to have a station number of 2. If you select No, the network conversion will be cancelled.

   Note: The Unsolicited_Mesgs node will automatically be removed when the application is converted to ControlNet.

   For details on configuring an application for ControlNet networks, see "Defining Communications for a Data Highway Plus or ControlNet Application," earlier in this chapter.

6. After converting the application to ControlNet, you need to change the Terminal Network Address on the Terminal Communications tab of the Configure Communication Setup dialog.

   The Terminal Network Address remains unchanged by the conversion. Ensure this value is still correct for your application, or change it. The range is 1 to 99 (decimal).

   Important: For versions 1.0 and 1.25 of ControlNet, the PanelView Terminal Network Address should not be set to 1.

   For example, if the Terminal Network Address for your DH+ application was 25 (octal), it will be 25 (decimal) for your ControlNet application.

   In the Network File Transfer tab, the PanelView Location Routing Address will be set to the Terminal Network Address.

Converting from ControlNet to Remote I/O

To change the network type from ControlNet to Remote I/O:

If you change an application from ControlNet to Remote I/O, you must define the necessary racks, block transfer files, and Pass-Through File control byte (if needed), as well as change any tag addresses that are not valid for Remote I/O. You will also lose your defined nodes and scan classes.

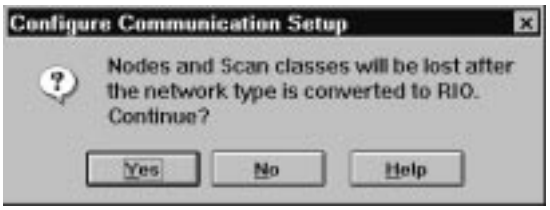
1. Choose Change Network Type from the Tools menu.
2. In the Target Network field, choose Remote I/O.
3. To convert the tags in the application, select the Convert Tags check box. If you don't want to convert tags in the application, clear the check from the Convert Tags check box.
   - If you don't convert the tags to Remote I/O format, the tags' data source remains Device and the address is unchanged. You will have to change the data source to Remote I/O manually for any of these tags that you want to use, and give them each a Remote I/O address.
   - If you convert the tags to Remote I/O, the data source for all the tags is changed to Remote I/O, but you will still have to give them valid Remote I/O addresses manually.

▶ **Note**: Converting shared tags may invalidate them for other applications that use them. Any unused tags will remain unconverted.

▶ **Note:** If you convert an application from Remote I/O to DH+ or ControlNet, or vice-versa, the tag addresses may be invalid for the network to which you are converting. Change the tag addresses to match the network type.

4. Choose OK.

   The following message will appear:


Configure Communication Setup
Nodes and Scan classes will be lost after the network type is converted to RIO. Continue?
Yes
No
Help

5. Choose Yes to confirm that you want to continue.

6. Once you have changed the network type you will have to configure the programmable controller type, rack assignments, block transfer files, and Pass-Through File control byte in the Rack and Block Transfer File editors. You may also have to re-specify the PanelView Location Routing Address on the Network File Transfer tab in the Configure Communication Setup dialog box.

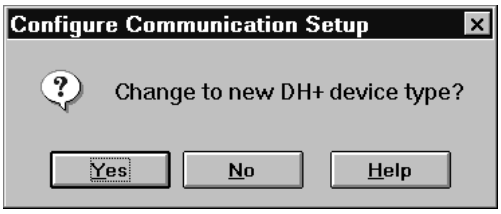
For details on configuring an application for Remote I/O, see "Defining Communications for a Remote I/O Application," earlier in this chapter.

Converting from ControlNet to DH+

To change the network type from ControlNet to DH+:

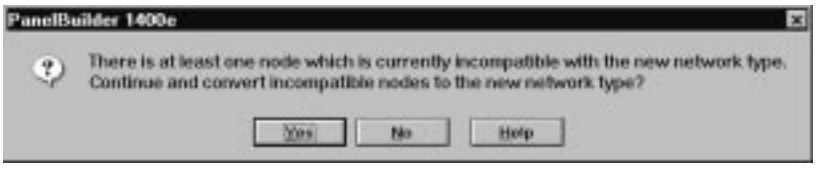
1. Choose Change Network Type from the Tools menu.
2. In the Target Network field, choose DH+.
3. Choose OK.

   The following message will appear:



4. Choose Yes to change your ControlNet application to DH+. Choose No if you do not wish to continue with the conversion.

   PanelBuilder checks that the station numbers of all ControlNet nodes will still be valid when the network type is converted to DH+. A message appears if invalid station numbers are detected. For example, station number 8 is an invalid octal number. The following message will appear, prompting you to convert invalid station numbers.


PanelBuilder 1400e
There is at least one node which is currently incompatible with the new network type. Continue and convert incompatible nodes to the new network type?
Yes
No
Help

Choose Yes to continue with the network conversion, or choose No to stop the conversion. If you continue, any invalid station numbers will be changed to a default value of 0.

For details on configuring an application for DH+, see “Defining Communications for a Data Highway Plus or ControlNet Application,” earlier in this chapter.

5. After converting the application to DH+, you may need to change the Terminal Network Address on the Terminal Communications tab of the Configure Communication Setup dialog.

The Terminal Network Address remains unchanged by the conversion. Ensure this value is still correct for your application on the network you are switching to; if not, change it. The range is 0 to 77 (octal).

For example, if the Terminal Network Address for your ControlNet application was 25 (decimal), it will be 25 (octal) for your DH+ application.

In the Network File Transfer tab, the PanelView Location Routing Address will be set to the Terminal Network Address.

Defining Tags

This chapter describes:

- tags and the tag database
- the importance of organizing tags
- creating tags in the Tag Database editor
- using the Tag Usage Viewer
- sharing a tag database
- importing and exporting a tag database

Important: Plan the tag database before creating tags. Using naming conventions and organizing tags with folders reduces the time required to build the tag database and edit it later. See Chapter 2, *Planning Applications*.

What Are Tags?

Tags offer these advantages:

- They let you associate a symbolic name (that is, the tag's name) with a PLC address.
- Tag names eliminate the need to duplicate PLC addresses. Once you define a tag, you can specify that tag anywhere you want to reference the address.
- You can easily change all instances of an address by just editing the tag.
- If you are creating more than one application that will be using the same PLC addresses, you can create one tag database and share it between the different applications.
- You can share tag information with other PanelBuilder 1400e and RSView applications, and import tags from other Allen-Bradley and Rockwell Software programs.

A tag database can be directly shared by PanelBuilder 1400e and RSView. However, since each product has different feature sets, product-specific information is stored with each tag. This product-specific tag information does not affect how the other product functions, but will show up as disabled options and fields in the Tag Database editor.

Tag Types

PanelBuilder uses the following types of tags:

This tag type:	Stores this type of data:
Analog	Range of values (depends on the data type (integer or floating point) selected) These tags can represent complex states such as temperature or the position of rotary controls.
Digital	0 or 1 These tags can represent devices that can only be on (1) or off (0), such as push buttons.
String	ASCII string, series of characters or whole words (maximum of 82 characters) These tags are used only when defining ASCII Input and ASCII Display objects, and for sending alarm messages to the PLC.
Block	A contiguous bit stream of up to 1024 bits. A block tag is used strictly for defining the Alarm Triggers tag and Acknowledge to PLC tag when multiple simultaneous alarms need to be monitored. Use Block Tags when using bit-triggered alarms. Note that PanelBuilder Version 2 supports Device block tags as well as Remote I/O block tags.

Read and Write Tags

At runtime, PanelBuilder objects and control functions can read from or write to PLC addresses assigned to any of the tag types. The terms *read tag* and *write tag* will sometimes be used. A read tag is a tag with a PLC address that only needs to be read by PanelBuilder objects and functions. A write tag is a tag with a PLC address that can be written to by PanelBuilder objects and functions.

For a Remote I/O application, a read tag can use any Full or Listen-Only I/O address. A write tag must use a discrete Full Access address or the block transfer file address of a file with a control byte defined on a Full Access input module.

Note: You can use expressions to perform calculations and logical and conditional operations to manipulate the values of tags. See Chapter 7, *Creating Expressions*.

Data Sources

The data source for a tag is the physical location within a programmable controller that PanelView is to read a value from or write a value to. The data source for tags in a PanelBuilder application can be either Device or Remote I/O. Which one you select depends on whether you are creating tags for a DH+, ControlNet, or Remote I/O application.

The other two data sources you can see in the Tag Database editor, Memory and DDE, are used only by RSView.

Device

When you create tags for a DH+ or ControlNet application, select Device as the data source. This means that the data comes from a specific memory location in a programmable controller that is connected to the DH+ or ControlNet network.

For each tag you must specify a node name, a scan class, and the physical address within the programmable controller that the tag is to reference. The node name refers to the programmable controller's location on the DH+ or ControlNet network; the scan class determines how often the terminal is to check the value at this address. See Chapter 2, *Planning Applications*, for details on configuring nodes and scan classes.

Remote I/O

When you create tags for a Remote I/O application, select Remote I/O as the data source. This means that data comes from a specific memory location in the programmable controller that is linked directly to the PanelView terminal through the I/O rack. For each tag, you must identify the address within the controller's Remote I/O racks by specifying a rack or block transfer file address. See Chapter 2, *Planning Applications*, for details about rack and block transfer file configuration, and Appendix C, *Addressing Syntax*, for addressing details.

Organizing Tags

Organizing tags makes database creation and maintenance faster and simpler and helps optimize PanelView performance. To organize tags:

- Establish naming conventions.

  Naming conventions enable you to use wildcards most effectively when searching for and displaying tags during development and runtime.
- Use folders to group related tags.
- Avoid defining more than one tag for a particular address.

  Each tag used in the application requires memory when the application is run, without regard to the tag's address. Additionally, keeping the address-tag relationship unique lets you change all references to an address by simply editing the tag.

Naming Tags

Tag names can be up to 40 characters long. If you create a folder, the folder name becomes part of the tag name.

The tag name can contain the following characters:

- A to Z
- 0 to 9
- underscore (_) and dash (–)

To separate the folder name from the rest of the tag name, use a backslash (\). For example, tags in the folder called Pump would start with Pump\. The folder name and separator character count in the 40 character tag name limit.

You can use mixed case. Tag names preserve upper and lower case for readability but are not case-sensitive. For example, the tag name MixerValve1 is the same as mixervalve1, but is displayed as MixerValve1.

Using Folders to Group Tags

To organize tags, create a folder and include tags that are related. For greater organization, you can nest folders. For example, you can organize the tag database first by area, then by machines in the area and finally by devices in each machine, nesting the folders at each step. Tag names in the last folder might begin with something like this: Area1\Machine1\Pump\.

The Tag Database Editor

Use the Tag Database editor to create or edit tags and folders. The spreadsheet viewer portion of this editor allows you to see the database structure.

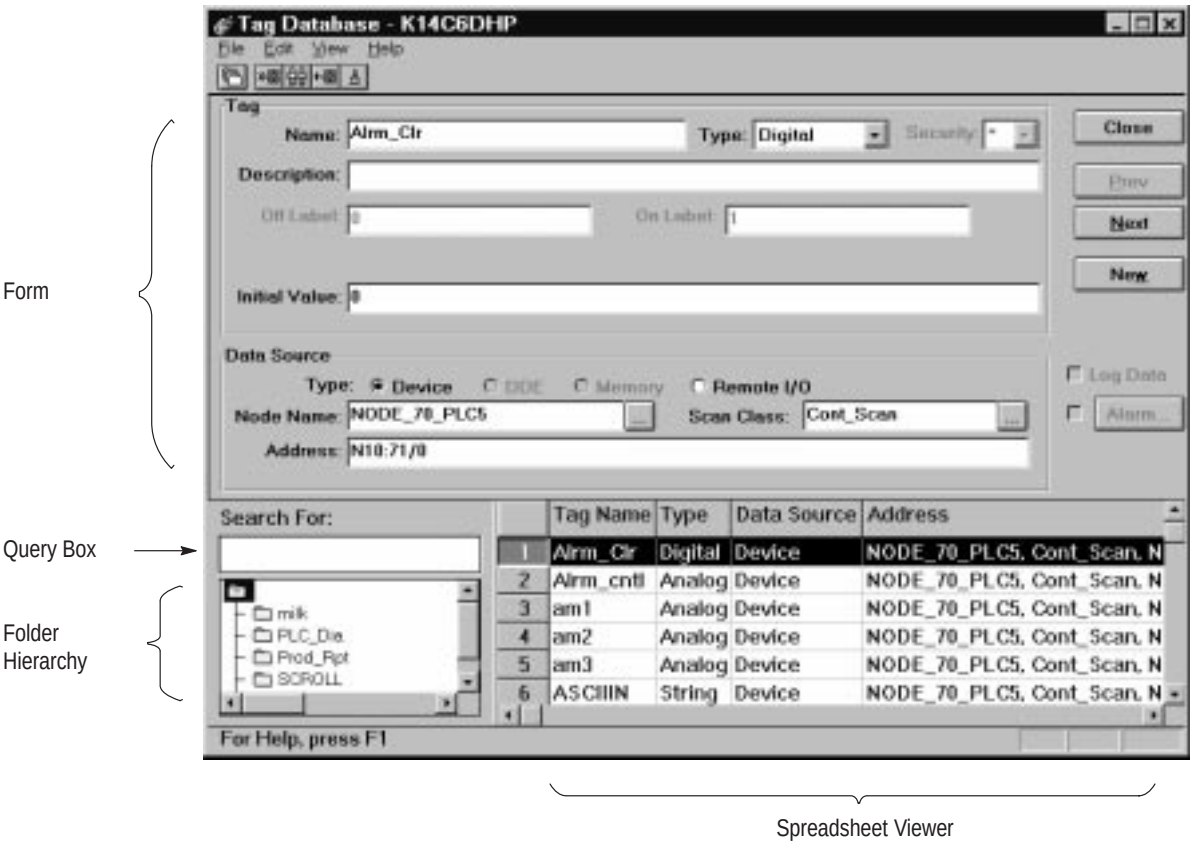
Note: The Tag Database editor works with both PanelBuilder and RSView databases, but, because the two have different product feature sets, some features will appear in the editor that are not relevant to editing tags in PanelBuilder. These features will be disabled.

Opening the Tag Database Editor

1. Open the application you want to edit.
2. Open the System folder and choose Tag Database, or choose from the toolbar.

Using the Editor

The Tag Database editor has the following parts: form, query box, folder hierarchy, and spreadsheet. Each part is described on the following pages. PanelBuilder allows both full and incomplete tag information to be saved. If all tag fields are filled in, only completely valid tag records are accepted. Otherwise, error messages flag the fields in error. If any of the data fields are not filled in, clicking the Accept button will save the incomplete tag record without validating it. This lets you define a tag first and fill in nodes, racks, scan classes and addresses later.



Using the Form

Use the form to create a tag. In the upper part of the form, define the basic tag characteristics, such as tag name, type, and specifics related to the tag type. In the lower part of the form, define the data source (where the tag's values will come from).

Using the Buttons in the Form

When you go into the Tag Database editor, the buttons in the form are Close, Prev, Next, and New. Close closes the editor. Prev displays the previous tag in the form section. Next displays the next tag. New clears the form input fields so you can define a new tag.

When you enter information, the Prev and Next buttons change to Accept and Discard. Choose Accept to save tag information and Discard to cancel changes to a tag.

Important: For shared databases, see "Using Shared Tag Databases," later in this chapter. Accepted changes are saved permanently to disk. For private databases, changes are only saved temporarily. You must save the PanelBuilder application to save the tag changes permanently.

Using the Query Box

Use the query box to select the tags you want displayed in the spreadsheet. You have two options:

- Select a single tag by typing the tag name.
- Select multiple tags by typing wildcards.

There are two wildcards:

This character:	Does this:
?	Matches any single character.
*	Matches any number of characters, including the backslash (\) character. Use this wildcard by itself to display all the tags in the tag database.

Press ENTER to display the results of your query in the spreadsheet viewer. The first tag that matches the query is displayed in the form.

Using the Folder Hierarchy

The hierarchy and spreadsheet work together. The hierarchy shows the tag folders and the spreadsheet shows the tags within the folders.

A folder icon called root is always present in the hierarchy window. It is at the top of the folder hierarchy.

Initially, the root folder is empty. You can add other folders to the hierarchy or you can nest folders within another. If a folder icon has a plus (+) sign, it contains one or more folders. If a folder icon is blank, it does not contain any other folders.

This folder contains one or more folders.

This folder does not contain another folder.

To view a series of nested folders or long tag names, resize the query and hierarchy boxes. Place the cursor over the right border of the box surrounding them. When the cursor changes to a double arrow, drag the border to the desired size.

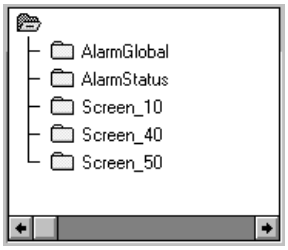
Tip: Before you begin defining tags consult your Device Tags or Remote I/O Tags Worksheets, and check whether you want to change or add to the folder hierarchy you planned in the worksheets.

Creating a Folder

1. Choose New Folder from the Edit menu.

2. Type the folder name.
3. Choose OK.

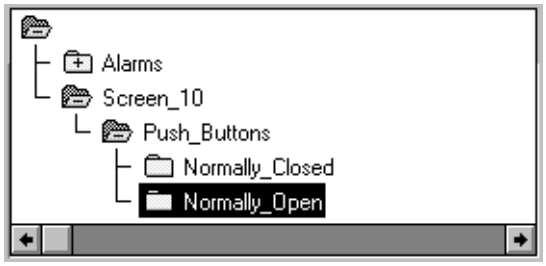
   The new folder appears in the hierarchy window.



Nesting Folders

There are two ways to nest a folder within another:

- Open the containing folder. Then add a new folder by specifying the folder name. The path of the containing folder will already be placed in the New Folder Name field.
- With all folders closed, add a new folder by typing its full path. For example, Screen_10\Push_Buttons\Normally_Open would add the folder highlighted in the illustration below.



Important: The maximum number of characters allowed in the folder path and tag name is 40. The backslash in the folder path counts as a character. Be aware of this when you are nesting folders. In the example above, the folder path takes up 36 characters. This leaves only 4 for the tag name (including a backslash).

Opening a Folder

When you open a folder, its contents are displayed in the spreadsheet.

To open a folder, either:

- double-click the folder
- select the folder and press Enter

To open multiple folders:

1. Use one of these methods to select the folders:
   - click a folder and drag the mouse up or down
   - click a folder and Shift-click other folders immediately above or below the first selected folder
   - click a folder and Ctrl-click other folders anywhere in the hierarchy
2. Press Enter.

   The tags in the selected folders are displayed in the spreadsheet.

Adding Tags to a Folder

Once you've created a folder, you can add tags to it:

1. Select a folder in the folder hierarchy. The folder name is displayed in the Name field of the form.
2. After the backslash (\), type the new tag name.

Note: To add a tag to another folder without closing the current folder, type the name of the tag with the full folder path.

Duplicating a Folder

When you duplicate a folder, the folder and any folders or tags within it are also duplicated. Make sure that you do not exceed the 40 character limit for folder path and tag name when you duplicate nested folders.

To duplicate a folder and its tags:

1. Select the folder in the hierarchy.
2. Choose Duplicate Folder from the Edit menu.

Important: Be sure to choose Duplicate Folder rather than Duplicate. Duplicate only duplicates individual tags.

3. Type a name for the new folder.
4. Choose OK.

Deleting a Folder

1. Select one or more folders in the hierarchy.
2. Choose Delete Folder from the Edit menu.

   The folder and all folders and tags within it will be deleted.

Using the Spreadsheet Viewer

Use the spreadsheet viewer to view tags or to select a tag to edit. Use the query box or folder hierarchy to select the tags you want to view.

Note: The Address column in the spreadsheet viewer contains node and scan class information for Device tags. The node name, scan class name, and address are separated by commas.

Moving Through the Spreadsheet

To scroll through the spreadsheet rows, either:

- use the Prev or Next buttons in the form area

  or

- use the Up and Down arrow keys

  or

- use the scroll bars in the spreadsheet viewers

To select a row, click anywhere in the row or click the row number.

Resizing Columns and Rows

1. Place the cursor over the division between the column or row until the cursor changes to a double arrow.
2. Drag the column or row to the desired size.

Adding a Tag

To insert a new row above the highlighted row, use one of these methods:

- choose the New button in the forms area

  or

- choose Insert Row on the Edit menu

  or

- choose the Insert Row button on the toolbar

Insert Row

Choose Accept to save the information for a tag, or Discard to discard it.

Note: The row position of the new tag is only temporary. Even if a tag is not displayed, it is still saved. When the spreadsheet viewer is refreshed, the new tag, if it still matches the query set, will be displayed in alphanumeric order relative to other tags in the database. Scrolling the spreadsheet, closing folders, and opening folders will cause a refresh.

Duplicating a Tag

Duplicate

1. Select the tag you want to duplicate.
2. Choose Duplicate on the Edit menu, or select the Duplicate button on the toolbar.

   A new row is inserted below the highlighted row. The new row contains all the same information except the tag name.
3. In the Tag Name field, type the name for the new tag.
4. Choose Accept to save the new row or Discard to cancel it.

Note: As with saves, the row position of the new tag is only temporary. Even if a tag is not displayed it is still duplicated. When the spreadsheet view is refreshed, if the duplicate tag still matches the query set, it will be displayed in alphanumeric order relative to other tags in the database. Scrolling the spreadsheet, closing folders, and opening folders will cause a refresh.

Editing a Tag

1. Select the tag you want to edit.

   The details of the tag appear in the form area.
2. Edit any details.
3. Choose Accept to save the editing changes, or Discard to cancel the changes.

Renaming a Tag

For information on renaming a tag, see “Renaming a Tag,” later in this chapter.

Deleting a Tag

Delete

1. Select the tag you want to delete.
2. Choose Delete on the Edit menu, or choose the Delete button on the toolbar.

Note: If you delete a tag that is assigned to an object, you will get a validation error. Use the Tag Usage Viewer to see which tags have been assigned to objects. See “Using the Tag Usage Viewer” later in this chapter for more information.

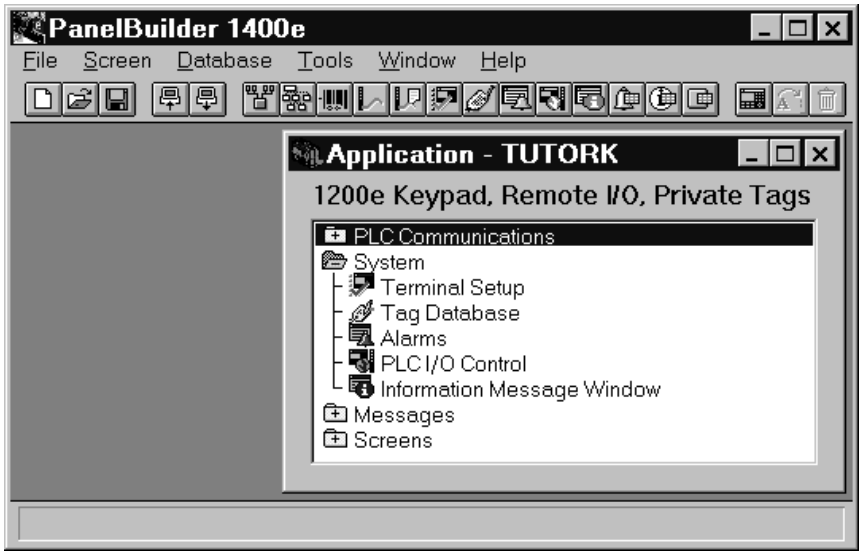
Working in Multiple Editors

As you work in the Tag Database editor you can open other editors to work in multiple editors simultaneously. For example, as you are creating tags for a DH+ or ControlNet application, you can add or edit a node in the Node editor.

To open another editor:

1. Click or choose Application Window from the View menu.

   The PanelBuilder window appears.



2. Click the name of the editor you want to open in the Application window, or choose its icon from the toolbar.
3. Press ALT-TAB to move between the open editors.

Note: If you make a change in the Scan Class editor while the Tag Database editor is open, when you return to the Tag Database editor the list of scan classes is automatically updated.

Configuring Tag Type

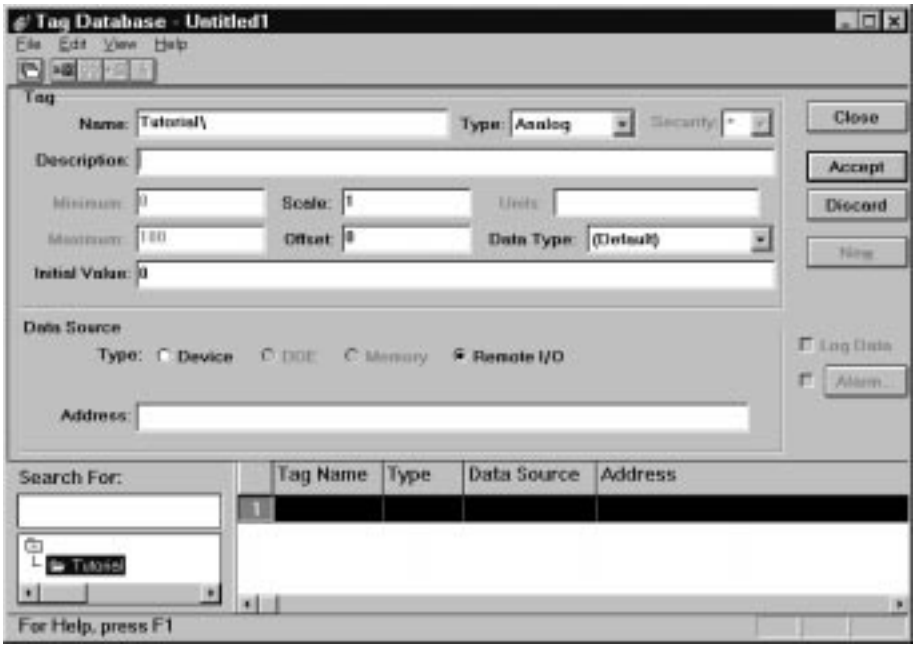
This section describes how to fill in the fields for analog, digital, string, and block tag types. Use the information you entered on the Device Tags and Remote I/O Tags Worksheets.

Note: The Log Data and Alarm check boxes in the Tag Database editor are present to maintain compatibility with RSView databases. You cannot use them in PanelBuilder.

Tip: For information about assigning tag addresses that optimize DH+ highway traffic, see "Guidelines For Organizing Tags to Optimize DH+ Network Performance" in Chapter 2, *Planning Applications.*

Configuring an Analog Tag

1. If the tag is part of a folder, select that folder in the folder hierarchy. The folder name appears in the Name field and is the first part of the tag name.
2. Type a tag name. If the tag is part of a folder, type the name after the backslash (\).



3. For Type, select Analog.
4. Fill in the fields as outlined below:

   Security
   This field is present to maintain compatibility with RSView databases. You cannot enter anything in the Security field.

Description
Type a description of this tag up to 128 characters long.

Minimum and Maximum
These fields are present to maintain compatibility with RSView databases. You cannot enter anything in the Minimum and Maximum fields.

Scale and Offset
Type a number. For the scale, do not use 0.

Scale and offset together transform raw PLC data into meaningful units for the operator. At runtime, all data read from the programmable controller into an analog tag will be scaled using the formula:

$y = (mx) + b$

where m = scale
x = raw PLC data
b = offset
y = scaled value

The terminal displays the scaled value. When data is written to the programmable controller using an analog tag, the value will be inversely scaled using the formula:

$$x = \frac{(y - b)}{m}$$

This converts the value back to raw PLC data before it is written to the programmable controller. With Scale = 1 and Offset = 0 (the default values for a new analog tag), programmable controller data is not changed.

Important: A fractional scale or offset value will result in a floating point scaled value. This will be true regardless of the tag's data type. Extensive use of floating point values can slow a terminal's performance.

Important: If the descaled value to be written to the tag address is a fraction, but the tag is not configured to deal with floating point data (i.e., Float data type or Default data type with a tag address in the Float section of the programmable controller), the value will be truncated to the nearest whole number before it is written to the tag address.

Note: You can also use expressions to perform calculations on the values of analog and digital tags that read information from the PLC. See Chapter 7, *Creating Expressions.*

Units
This field is present to maintain compatibility with RSView databases. You cannot enter anything in the Units field.

Data Type
Select the data type that matches the format of the data stored in the programmable controller. The default data type is Default. In DH+ and ControlNet applications, Default will let the tag address dictate the tag's data type (for example, N7: - integer; H3:0 - long). Unless special data is expected, Default should be acceptable. For Remote I/O applications, Default means Unsigned Integer. You will need to set the data type to the format of the data expected in the I/O tables.

Remote I/O

For Remote I/O tags, the address length depends upon the data type assigned. The default data type is Unsigned Integer with an address length of 16 bits. When assigning a Binary or Bit Position data type, unless an address length is specified, the address length defaults to 1 bit. Byte defaults to 8 bits, Long Integer defaults to 32 bits, and 1 to 8 BCD defaults to 4 to 32 bits.

You can also use the Floating Point data type for Remote I/O applications. Tags that use this data type must use a block transfer file address.

Note: If you use the Floating Point data type for a Remote I/O application, the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Note: The BIN data types are reserved for Modbus applications, and cannot be used for DH+, ControlNet, or Remote I/O applications. For more information about creating applications for Modbus, see the *PanelBuilder 1400e Configuration Software for Windows Modbus User Manual*, Catalog Number 2711E-6.12.

DH+ and ControlNet

For DH+ and ControlNet Device tags, the number of bits in an address is always determined by the address's data source. No address length is required. Addressing a float section of the PLC controller allows the terminal to access float values.

Initial Value
This is the default value of the tag at the terminal. When a tag is used with a PanelBuilder input object or function, this initial value, after descaling, is written to the programmable controller when the application first executes.

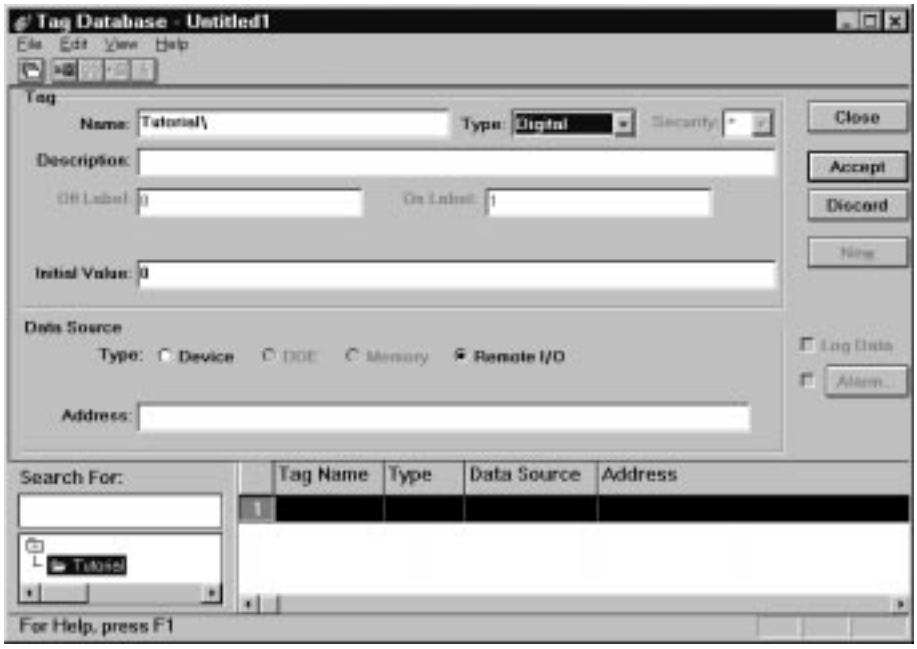
For a PanelBuilder object with an initial state and a main control tag, the object's initial state value supersedes the tag's initial value, so that the initial state value is written to the programmable controller instead of the tag's initial value. On application upload, the object's initial state value is used as the tag's initial value.

5. To fill in the fields for data source, see "Specifying a Data Source," later in this chapter.

Configuring a Digital Tag

A digital tag has two possible values at runtime: 0 or 1. Use a digital tag wherever a toggle or Boolean function is needed.

1. If the tag is part of a folder, select the folder in the folder hierarchy. The folder name appears in the Name field and is the first part of the tag name.
2. Type a tag name. If the tag is part of a folder, type the name after the backslash (\).
3. For Type, select Digital.



4. Fill in the fields as outlined below:

Security
This field is present to maintain compatibility with RSView databases. You cannot enter anything in the Security field.

Description
Type a description of this tag up to 128 characters long.

Off Label and On Label
These fields are present to maintain compatibility with RSView databases. You cannot enter anything in them.

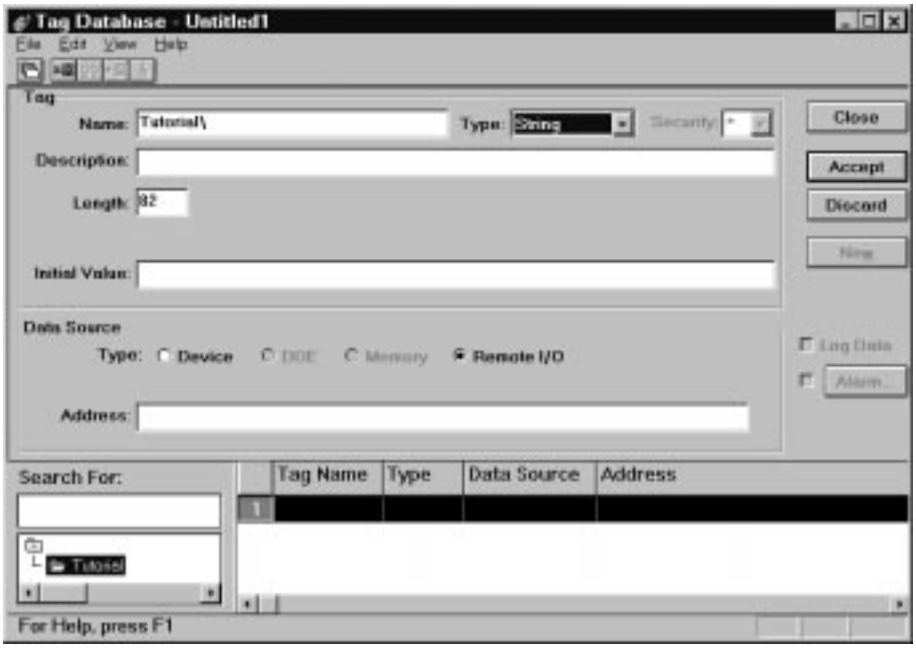
Initial Value
Enter an initial value of 0 or 1 for the digital tag. When a tag is used with a PanelBuilder input object or function, this initial value is written to the programmable controller when the application first executes.

For a PanelBuilder object with an initial state and a main control tag, the object's initial state value supersedes the tag's initial value, so that the initial state value is written to the programmable controller instead of the tag's initial value. On application upload, the object's initial state value is used as the tag's initial value.

5. To fill in the fields for data source, see "Specifying a Data Source," later in this chapter.

Configuring a String Tag

1. If the tag is part of a folder, select the folder in the folder hierarchy. The folder name appears in the Name field and is the first part of the tag name.



2. Type a tag name. If the tag is part of a folder, type the name after the backslash (\).
3. For Type, select String.

4. Fill in the fields as outlined below:

   Security
   This field is present to maintain compatibility with RSView databases. You cannot enter anything in the Security field.

   Description
   Type a description of this tag up to 128 characters long.

   Length
   Type a number between 1 and 82 to specify the length of the string tag in bytes. The length must be a multiple of the size of the programmable controller data element you're addressing.

   Important: For device tags, odd lengths are permitted in the string section only. For Remote I/O tags, string lengths must be a multiple of 2.

   Initial Value
   Enter an initial value for the tag. When a tag is used with a PanelBuilder input object or function, this initial value is written to the programmable controller when the application first executes.

   ▶ **Note:** There can be no more than 720 string tags in an application.

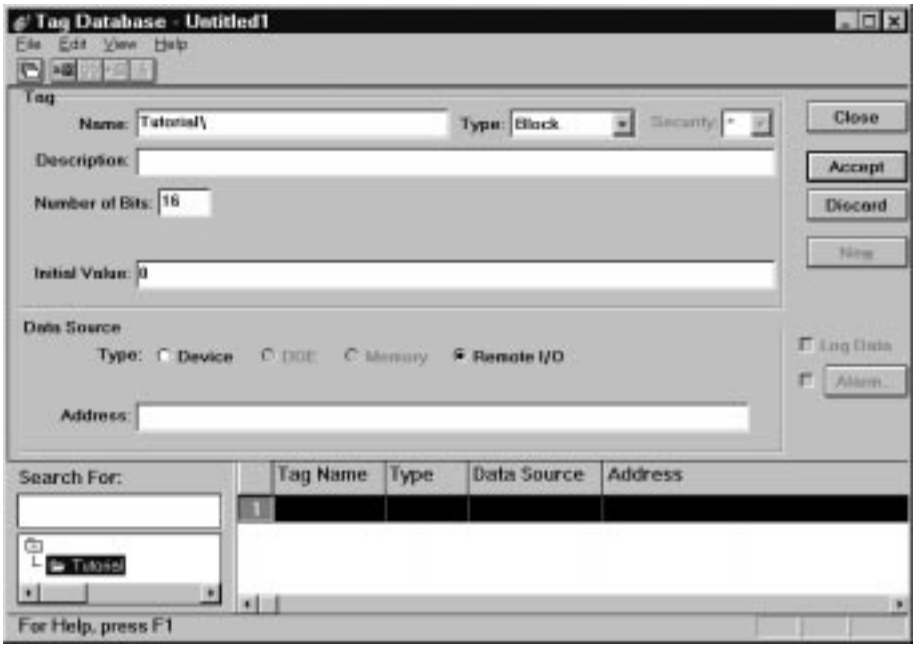
5. To fill in the fields for data source, see "Specifying a Data Source," later in this chapter.

Configuring a Block Tag

Block tags can be defined in a Remote I/O application or in a DH+ or ControlNet application. If you use block tags in a DH+ or ControlNet application, the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Important: Block tags are used only for the Alarm Control and the Alarm Ack Value control when multiple, simultaneous alarms need to be monitored. For more information, see Chapter 11, *Configuring Alarms*.

1. If the tag is part of a folder, select the folder in the folder hierarchy. The folder name appears in the Name field and is the first part of the tag name.
2. Type a tag name. If the tag is part of a folder, type the name after the backslash (\).
3. For Type, select Block.



4. Fill in the fields as outlined below:

   Security
   This field is present to maintain compatibility with RSView databases. You cannot enter anything in the Security field.

   Description
   Type a description of this tag up to 128 characters long.

   Number of Bits
   Enter a number from 1 to 1024 to define the length of the block.

   RIO
   For Remote I/O tags, discrete addresses can accommodate 128 bits (8 words at 16 bits/word). Use a block transfer file address for longer block lengths.

   DH+ and ControlNet
   For Device tags, the number of bits must correspond to the element in the address. For example, if the address is located at an integer data file element, the length must be a multiple of 16, the bit length of an integer.

   Initial Value
   Enter a 1 or 0 to initialize the entire block to 1 or 0 (that is, all bits 1 or all bits 0). When a tag is used with a PanelBuilder input object or function, this initial value is written to the programmable controller when the application first executes.

5. To fill in the fields for data source, see “Specifying a Data Source,” next.

Specifying a Data Source

The data source determines where a tag's data comes from. Tags defined for a Remote I/O application must specify the Remote I/O data source. Tags defined for a DH+ or ControlNet application must specify the Device data source.

Note: DDE and Memory data sources are only used by RSView.

For a Remote I/O application, a tag that is going to write to the programmable controller must address an input module in a rack that has Full access. If it is only going to read from the programmable controller, it can address an input or output module in a rack that has Full or Listen Only access.

For a DH+ or ControlNet application, the ladder logic program in the programmable controller determines whether or not it is appropriate for the terminal to read from or write to the PLC.

Important: For DH+ and ControlNet analog tags, the entire PLC data file element assigned to the tag's address is used in determining the tag's value. This is true regardless of the tag's data type. For example, a Byte data type assigned with the address N7:0 will still look at the entire 16 bits of the integer element, even though the tag is interested only in the lower 8 bits. PLC logic must ensure that the unused bits are 0, to prevent invalid or out of range data from being read from the address.

Device Data Source

A tag with Device as its data source receives its data from a programmable logic controller on the DH+ or ControlNet network. If you have not configured the node and scan class for the tag, you can configure them from the Tag Database editor while you are defining a tag, or you can configure them after you have finished creating tags. If you are going to define them after the tag is created, make sure one of the node, scan class, or address fields is blank so that when you choose Accept, PanelBuilder will not validate the tag and expect a valid tag record complete with supporting node and scan class definitions. See "Relaxed Validation," below.

Remote I/O Data Source

A tag with Remote I/O as its data source receives its data from a programmable logic controller linked directly to the PanelView terminal through the I/O rack. If you are using discrete addresses, you must configure the Remote I/O racks before you can assign an address to a tag. If you are using block transfer files, you must configure these in addition to the racks, before you can assign an address to a tag.

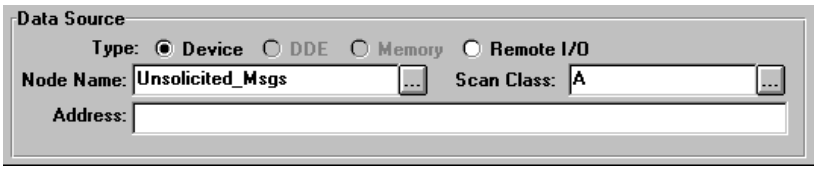
Relaxed Validation

When you configure the data source for a tag, if you leave any of the fields in the Data Source form section blank, the tag definition will be saved but not validated. If all the fields are filled in, the tag will be validated when you choose Accept, and any errors must be corrected before the tag definition will be saved. Validation will include ensuring that any node, scan class, rack, and block transfer file definitions needed by the tag exist.

Specifying Device as the Data Source

Use the information on the Device Tags Worksheets to fill in the tag addresses.

1. Choose Device.



The screen defaults to Node Name: Unsolicited_Msgs and Scan Class: A and you must make the changes as outlined in the next step.

2. The node is the programmable controller this tag addresses. In the Node Name field, do one of the following:
   - Type a node name.
   - Double-click in the Node Name field to open the Add/Edit Node dialog box.

   If the field is empty you can create a new node definition; if the field contains the name of an existing node, you can edit that node's definition.

   - Choose ... to open a selection list and select a node name.

   For information about configuring nodes see Chapter 5, *Defining Communications*.

3. Select a scan class.

   For information about configuring scan classes, see Chapter 5, *Defining Communications*.

4. In the Address field, type the tag's physical memory location in the PLC.

   Note: For Device block tags, the address is the starting address for the block. The address must start on an element boundary (sub-element starting addresses are not supported). The address cannot cross PLC data file boundaries.

   The address syntax depends on the programmable controller's addressing convention. For detailed information about address syntax, see Appendix C, *Addressing Syntax.*

Specifying Remote I/O as the Data Source

Use the information on the Remote I/O Tags worksheets to fill in the tag addresses.

1. Choose Remote I/O.
2. In the Address field, type the address.

Note: For analog tags, the address length depends upon the data type assigned. The default data type is Unsigned Integer, with an address length of 16 bits. When assigning a Binary or Bit Position data type, unless an address length is specified, the address length defaults to 1 bit. Byte defaults to 8 bits, Long Integer defaults to 32 bits, and 1 to 8 BCD defaults to 4 to 32 bits. When assigning a Floating Point data type, you must use a block transfer address.

The address syntax depends on the type of address you are defining, discrete or block transfer. For detailed information about address syntax, see Appendix C, *Addressing Syntax.*

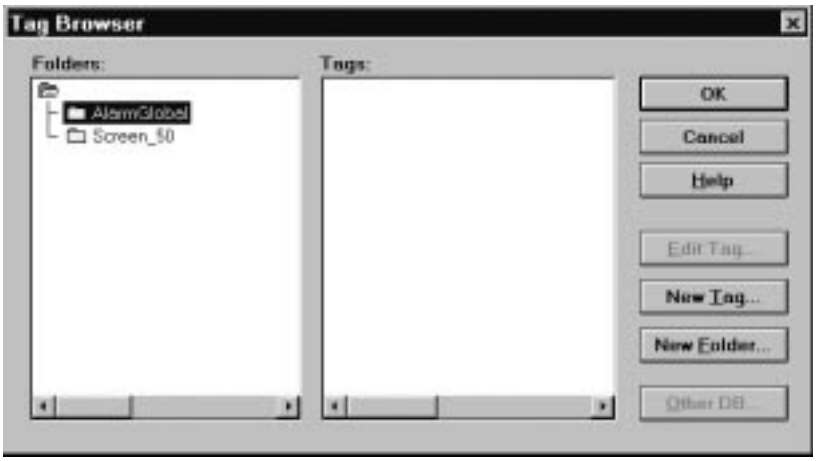
Other Methods of Creating Tags

If you need to create a large number of tags, you can use a third-party spreadsheet such as Microsoft Excel to do so. Then you can import the tags to the application using PanelBuilder's Database Import function. You can also create tags as you need them in other PanelBuilder editors.

Creating Tags as Needed

You can add tags in any editor that can assign tags, such as the PLC I/O Control editor. Select the Assign Tag button to bring up the Tag Browser dialog box, which lets you browse the tag database, add a new folder, or bring up the Tag Editor dialog box to add a new tag or edit an existing tag.

- Highlight the tag field and click the Assign Tag button. The Tag Browser dialog box opens.



In this dialog box, you can perform these actions:

OK

- Select a tag by double-clicking it, or by highlighting the desired tag and clicking OK. The tag is assigned to the input field in the original dialog box.

Edit Tag...

- Edit a tag by highlighting it and then choosing Edit Tag. The Tag Editor dialog box appears, so that you can edit the tag's definition.

New Tag...

- Create a new tag by opening the desired folder and choosing New Tag. The Tag Editor dialog box appears, so that you can define the tag.

New Folder...

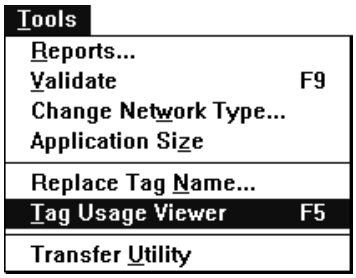
- Create a new folder by clicking the New Folder button, then typing the folder name in the New Folder dialog box.

The Tag Browser dialog box also has a button labeled Other DB. This button is dimmed because this feature is not active. It is there for compatibility with RSView databases.

For details on how to define tags, see "Configuring Tag Type," earlier in this chapter.

Note: If the Tag Database editor and Tag editor dialog boxes are open at the same time, when you add a tag in the Tag editor, it automatically appears in the Tag Database editor. When you add a tag in the Tag Database editor, to see the change in the Tag editor, double-click the folder to which the tag was added. In the Tag Database editor, you can also click the Refresh icon, or choose Refresh from the View menu, to update the display and sort the tags.

Using the Tag Usage Viewer



The Tag Usage Viewer window provides you with a cross-reference listing of all the tags in an application sorted according to their address, and shows where each tag is used. This listing is similar to the Tag Address Cross Reference Report. For tags in a DH+ or ControlNet application, it also shows the node name and scan class.

Use the Tag Usage Viewer to help locate tag validation errors.

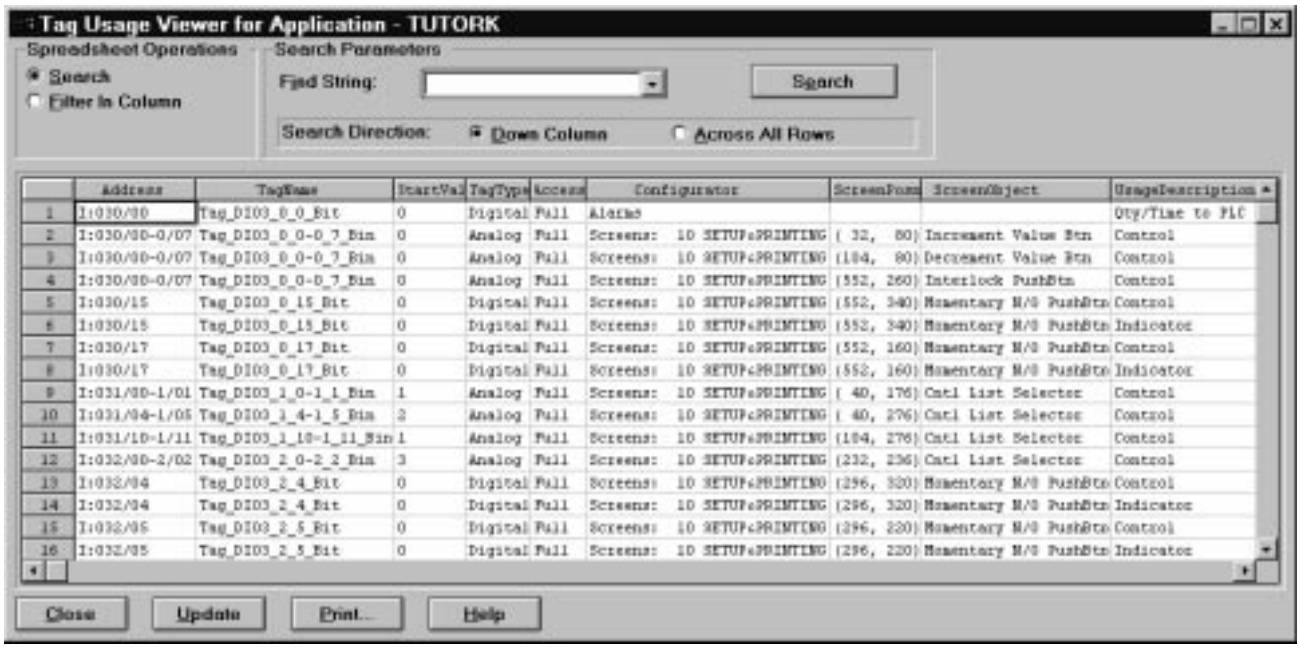
To open the Tag Usage Viewer:

- Choose Tag Usage Viewer from the Tools menu, or press F5. The system validates the screens and tags before it brings up the display.

To use the Tag Usage Viewer for Screens:

- When editing screens, choose Screen Tag Usage Viewer from the Screen menu, or press F5. PanelBuilder displays only the tags used on the current screen.

Note: Because the Tag Usage Viewer uses the Windows default printer setup, use the Windows Print Manager to change the printer settings (for example, to set it to Landscape mode so that all the columns will be printed).



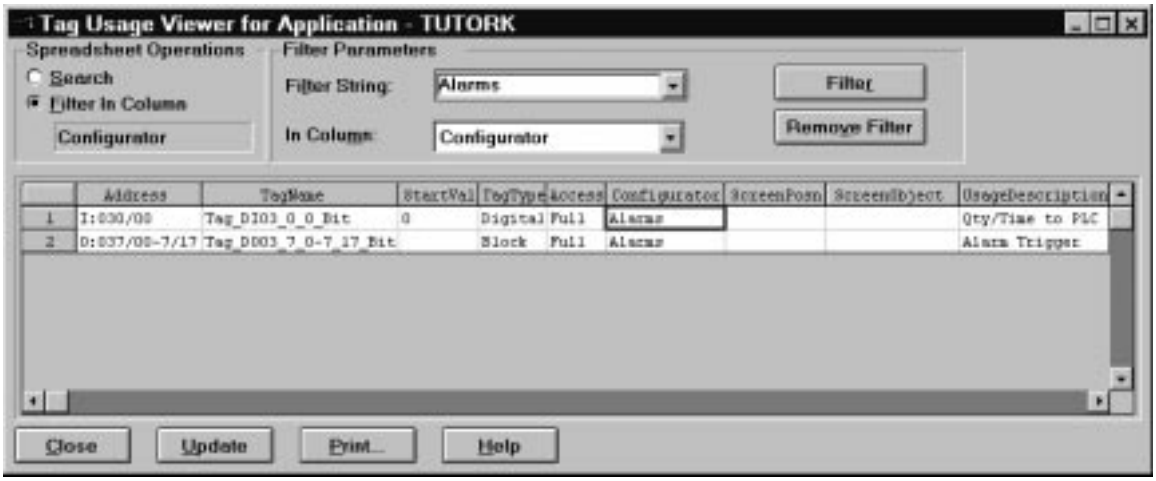
Initially, all tags used in the application (or on the screen) are listed in the Tag Usage Viewer. If you assign tags to expressions, those tags also are listed in the Tag Usage Viewer. You can restrict the list of tags by applying a filter, search for specific characters (such as an address or tag name) in the listing, and print out the listing.

Filtering Data

A filter is a way of screening the list of tags. Only those tags that match the filter criteria will be viewed.

Note: Filters remain in effect throughout other operations. If you set a second filter, you will be filtering only the filtered listing. Similarly, if you search or print, you will search or print only the filtered list of tags.

The last filter applied is shown beneath the Filter In Column button. In the example below, the last filter applied is a filter string on the Configurator column.



Tip: To see a list of tags associated with one particular screen, open that screen and choose Tag Usage Viewer from the screen editor's Screen menu.

To filter the list of tags:

1. Select the Filter In Column button.
2. Choose the column you want to set the filter in from the drop-down list in the In Column field.
3. Specify the limiting filter in the Filter String field. The filter is not case-sensitive.
4. Select the Filter button.

Example: Setting Filters

To set a filter to view tags that have addresses in the output file only, do this:

1. Select the Filter in Column button.
2. Choose Address from the drop-down list in the In Column field.
3. Type `O:` in the Filter String Field.
4. Select the Filter button.

To set a filter to view tags associated with the Alarm window only, do this:

1. Choose Configurator from the drop-down list in the In Column field.
2. Type
   `Alarms`
   in the Filter String column.
3. Select the Filter button.

All the tags associated with the Alarm window will be listed.

To remove a filter:

1. Select the Filter In Column button.
2. Select the Remove Filter button.

This removes not only the most recent filter, but all the filters that have been applied to the list of tags.

Searching for Text

You can search for any specific text in the Tag Usage Viewer.

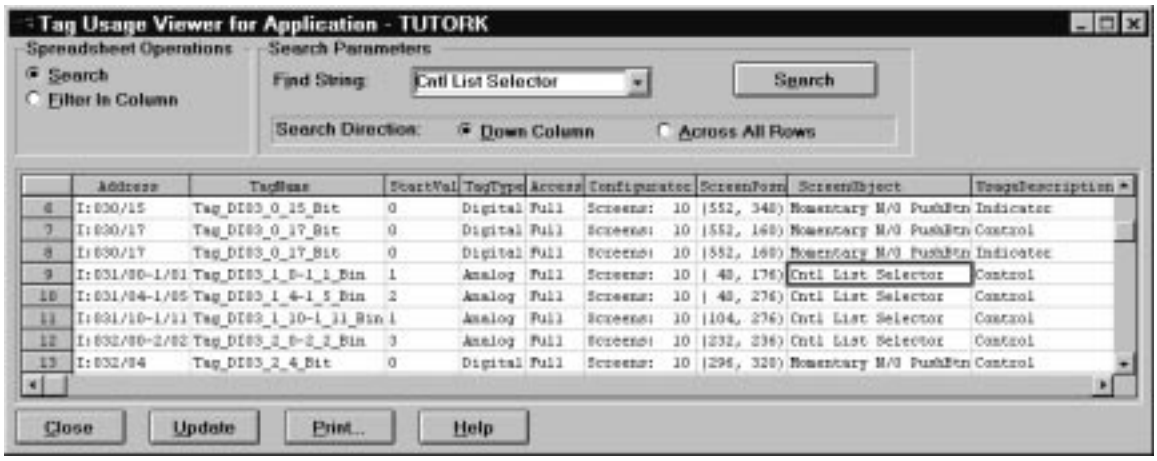
To search for text in the Tag Usage Viewer:

1. Select the Search button.
2. In the Find String field, specify the text you are searching for. The search is not case sensitive.
3. Specify whether you want to search down a particular column or across all rows by choosing Down Column or Across All Rows in the Search Direction field.

The Down Column option searches only in the currently selected column.

The Across All Rows option searches the entire list, starting at the top left corner, working its way across the row before moving on to the next row.

4. Select the Search button.


Tag Usage Viewer for Application - TUTORK
Spreadsheet Operations
Search
Filter In Column
Search Parameters
Find String:
Cntl List Selector
Search
Search Direction:
Down Column
Across All Rows
Address
TagName
StartVal
TagType
Access
Configurator
ScreenForm
ScreenObject
UsageDescription
6 I:030/15 Tag_DI03_0_15_Bit 0 Digital Full Screens: 10 (552, 348) Momentary N/O PushBtn Indicator
7 I:030/17 Tag_DI03_0_17_Bit 0 Digital Full Screens: 10 (552, 160) Momentary N/O PushBtn Control
8 I:030/17 Tag_DI03_0_17_Bit 0 Digital Full Screens: 10 (552, 160) Momentary N/O PushBtn Indicator
9 I:031/00-1/01 Tag_DI03_1_0-1_1_Bin 1 Analog Full Screens: 10 (48, 176) Cntl List Selector Control
10 I:031/04-1/05 Tag_DI03_1_4-1_5_Bin 2 Analog Full Screens: 10 (48, 276) Cntl List Selector Control
11 I:031/10-1/11 Tag_DI03_1_10-1_11_Bin 1 Analog Full Screens: 10 (104, 276) Cntl List Selector Control
12 I:032/00-2/02 Tag_DI03_2_0-2_2_Bin 3 Analog Full Screens: 10 (232, 236) Cntl List Selector Control
13 I:032/04 Tag_DI03_2_4_Bit 0 Digital Full Screens: 10 (296, 320) Momentary N/O PushBtn Control
Close
Update
Print...
Help

The field containing the first occurrence of the specified text is highlighted. In the example above, the first occurrence of the specified text, Cntl List Selector, is highlighted. Press the Search button again to continue the search.

When the search has reached the bottom of the list, it will wrap to the top and begin again.

You can search a filtered list. If one or more filters are in place, the most recent filter is named under the Filter In Column button.

For information on customizing the display in the viewer, see "Changing Column Width and Row Height," in Chapter 4, *Working with Editors*.

Note: The Tag Usage Viewer retains the new settings when you change column widths. The next time you open the Tag Usage Viewer, columns you shrank to hide will appear as thin strips that you can resize to hide or reveal. Rows are restored when you reopen the viewer.

Renaming a Tag

Use the following procedure to rename specified instances or all instances of a tag.

1. From the Application Window, choose Replace Tag Name on the Tools menu.
2. Type the tag name you want to change, or choose ... to select a tag name from the Tag Browser. Or, click the down arrow to choose a tag name from the four most recent entries in this field.
3. Type the name of a new or existing tag in the Replace With box.
4. Choose a Search In option:

   Whole Application—replaces all instances of the tag name specified in the Look For field with the tag name specified in the Replace With field.

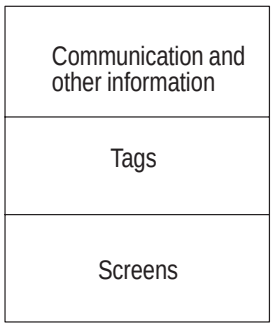
   Selection—replaces the tag name specified in the Look For field with the tag name specified in the Replace With field only in the selected program components.
5. Choose whether to rename the tag in the Tag Database automatically. By default, the Rename in Database box is checked. Uncheck it to allow you to change a tag name to one that doesn't exist in the Tag Database. You must create the tag manually before you validate the application.
6. Choose whether you want the application to be validated automatically once the tag has been renamed. By default, the Validate on Completion box is checked. Uncheck the box if you have unchecked the Rename in Database box.
7. Choose Replace.

Private and Shared Tag Databases

A PanelBuilder 1400e application can have its own private tag database or it can share a database with other PanelBuilder 1400e or RSView applications. This means if you have applications that require the same tags, you can define each tag once in one database, and share this database with other applications.

The Private Database

When you create a new application, its tag database is private, and is stored as part of the application file. All the information relating to the application, from the communication definitions (that is, nodes, scan classes, racks, and block transfer files) to the tags and the screens, is contained in one file. This file has the extension .pvc.



The application file (.pvc)

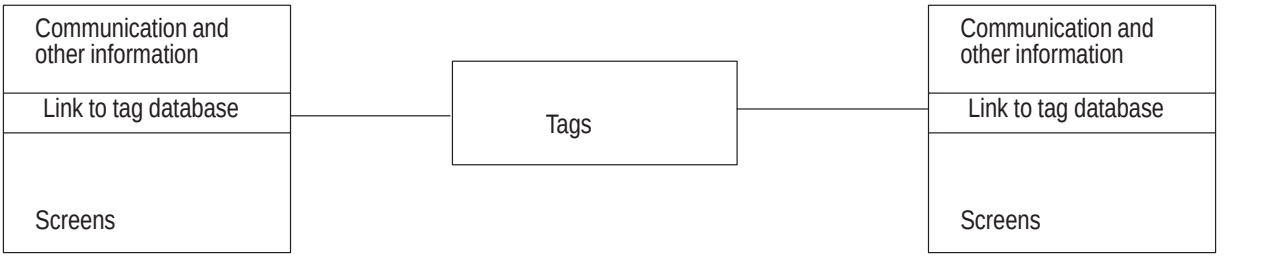
When a tag database is part of the application file, no other applications can access it. If you are creating a single application, or if none of the tags in your applications are common, this is all you need.

Note: An advantage of using a private tag database is that the application is self-contained within the .pvc file. This simplifies moving the application (complete with its tags) to another machine or directory.

Important: When you make changes to the tags in a private database, the changes are not saved until you save the application.

The Shared Database

If you are creating multiple PanelBuilder 1400e applications, or if you are working on a PanelBuilder 1400e application and an RSView project, you may want to share the tag database with more than one application. When you convert a private database to a shared database, the database is stored outside the application file. This improves the speed of opening and saving a large tag database.



A PanelBuilder 1400e Application | The Shared Database | Another PanelBuilder 1400e Application

Important: The communication information for each application is still private. However, when you select another PanelBuilder 1400e application to share its database with your current application, its communication information replaces the communications setup for your current application. Changes you make to the communication information once an application is using a shared database will not affect the information in the other application.

Using Shared Tag Databases

When you create a new application, its tag database is private. If you want the database to be shared by other applications, you must first convert the private database to a shared database. Converting to a shared database creates an external database with links from the application to the database.

Note: When you convert a private database to shared, the application's private database is deleted from the .pvc file.

Keep the following points in mind when sharing databases:

- A PanelBuilder 1400e application can only select a database from another PanelBuilder 1400e application of the same network type. However, a PanelBuilder 1400e application of any network type can select an RSView project.
- Each application using the database can modify the database, so make sure that any definitions you change won't adversely affect the other applications.
- Databases can be shared only by applications on the same physical computer. They cannot be shared over a network.
- When you select a PanelBuilder 1400e database for the current PanelBuilder 1400e application to share, you specify the PanelBuilder application, not the database itself. The source application's communication information (nodes, racks, and so on) is copied into the current application.

  Important: If an application already has communication information defined, it will be overwritten when you select a shared database.

- When you select an RSView database, no communication information is copied to the PanelBuilder 1400e application. You must define node and scan class information for the application using the Nodes and Scan Classes editors. RSView does not support Remote I/O tags.

- In order to select an RSView project, RSView must be installed on the same computer as PanelBuilder. If RSView is not installed on your machine, you need to export the RSView project as a .csv file, then import this file into PanelBuilder. You can also use the Repair Shared command on the Database menu to point to the RSView .db file.
- When you share a database between PanelBuilder 1400e and RSView, PanelBuilder 1400e cannot edit tags that it cannot use, such as RSView tags with Memory, System, or DDE as their data source.
- When you work with a shared database, your changes are saved permanently each time you select the Accept button.

Converting a Private Database to Shared

Convert a private database in either of these situations:

- You want the application to share tags created by another PanelBuilder 1400e application or by an RSView database.
- You want other PanelBuilder 1400e applications to share the current application's tags.

Note: When you move an application that is using a shared database to another machine, you must also move the database and use the same directory structure. If the database is not in its original location, use Repair Shared on the Database menu to reconnect the application to the tag database.

To convert a private database to shared:

1. Open the application that contains the database you want to convert to shared.
2. Choose Convert To Shared from the Database menu.
3. Fill in the fields as follows:

   Tag Database Name
   Displays the name of the current application, if it has been named. You can use this as the tag database name or type a new name.

   Tag Database Path
   Displays the path where the shared database will be placed. You can specify another path by selecting another directory, subdirectory, or drive, or use the default.
4. Click OK to complete the conversion.

Selecting a Shared Database

You can select another PanelBuilder 1400e database or an RSView database.

If you select another PanelBuilder 1400e database, the network type must be the same in both applications. The selected application's communication profile information—nodes and scan classes for DH+ and ControlNet applications, or racks and block transfer files for Remote I/O applications—is copied into the destination .pvc file. This overwrites the current application's communication profile and ensures that it will be ready to use tags in the shared database.

If you select an RSView database, only the tag definitions in the database are shared. You will have to configure node and scan class information for the current application.

Note: An RSView database can only be selected if RSView is installed on the same development computer as PanelBuilder 1400e. Sharing across a network is not supported.

To select a shared database:

1. Open the application for which you want to select a shared database.
2. Choose Select Shared from the Database menu.
3. Fill in the fields as outlined below:

   File Name
   Type the name of the application whose database file you want to share, or select the application from the list of files.

   List Files of Type
   Shows the type of files that can be shared by the current application. Select .pvc for PanelBuilder files and .rsv for RSView files.

   Directories
   Choose the directory containing the database you want to select.

   Drives
   If the directory containing the desired database is not displayed, you can select the drive where the directory is located.
4. Choose OK to share the specified database.

Repairing a Shared Database

You can repair the links to a shared database in these circumstances:

- The application has been moved to another machine where the database is in a different path.
- The database has been deleted and you have to link to another.
- The database has been corrupted and you have to link to another.

Note: The application's communication information is not affected when you repair a shared database.

To repair a shared database:

1. Open the application with the database you want to repair.
2. Choose Repair Shared from the Database menu.
3. Type the database name and path specification, or select the Browse button to search for a tag database. A database file will have the extension .db.
4. Choose OK to connect the application to the specified database.

Note: If you get the message "The file specified is not a tag database" when you try to repair a database, either you are pointing to the wrong file, or the current version of the database has been damaged beyond repair. Perhaps it has had some of its files deleted. If so, you will have to use a back up copy of the database.

Tag Database Import and Export

You can export a tag database created in PanelBuilder to a comma separated variable (.csv) file that can be used by third-party software such as Microsoft Excel. You can also create a database in Excel, save it as a .csv file, and then import it into PanelBuilder.

Tip: An exported PanelBuilder 1400e database can also be imported back into the application. This means you can use the Database Export feature to backup or update the database.

When you import a database, the tag definitions are copied into your current database. If the database is private, it remains private. A shared database remains shared.

PanelBuilder supports the following import file formats:

- PanelBuilder .csv—exported by RSView and PanelBuilder
- A.I. Series .csv—exported by the A.I. Series software
- Logic 5 .csv—exported from a Logic 5 database (see note below)
- PanelBuilder Tag Database—an independent tag database
- Taylor® ProWORXPLUS® database
- MODSOFT® database in flat text format

Note: To import information from an A.I. Series database, you must first export the database in text format using the A.I. software. PanelBuilder imports this text file. To import a Logic 5 database, export it to a text file and import this into A.I. software. You can then export the text file from A.I. as you would any other A.I. database, and import it to PanelBuilder. You can also import a PanelBuilder 900 database by exporting it in an A.I. text format, and then importing this into PanelBuilder as an A.I. database.

Note: To import or export a Taylor ProWORXPLUS database, or a MODSOFT database, see the *PanelBuilder 1400e Configuration Software for Windows Modbus User Manual*, Publication Number 2711E-6.12. This manual is supplied as part of the optional Modbus Communications Kit, Catalog Number 2711E-UMOD.

To import a database or .csv tag file:

1. Open the application to which you want to import a tag database.
2. Select Import/Export from the Database menu.
3. From the list that is displayed, choose the type of database to import.
4. If desired, select one of these options:
   - To make the import faster, select the Skip Existing Tags button.
   - To update existing tags from the tags being imported, select the Update Existing Tags button.
5. Select Next.
6. If you are importing a tag database, as opposed to a .csv format file, specify a Tag Database Source, or select the Down Arrow to display the list of sources, and choose one from it.

   If you are importing a .csv tag file or a text database, specify the file or choose the ... button and select the file from the Select File dialog box.
7. Select Next.
8. Select Finish.

To export a database:

1. Open the application from which you want to export a tag database.
2. Select Import/Export from the Database menu.
3. Select Export to PanelBuilder 1400e CSV Tag file and choose Next.

A default destination file name (pvtags.csv) and path are displayed.

4. Choose Finish to complete the export, or type a new destination file name and/or path, and then choose Finish.

Note: If you import a database that contains the Unsolicited_Msgs node into a ControlNet application, you must delete the node before you download the application. Otherwise, PanelBuilder will report a validation error.

CSV File Column Ordering

When you convert a PanelBuilder database to a comma separated variable (.csv) file, or when you create a spreadsheet in third-party software, it should consist of the following columns:

Column	Description
A Tag Type	A, D, S, T, B, or F (Analog, Digital, String, Structure, Block, Folder)
B Tag Name	Any legal tag name
C Tag Description	Text describing the tag
D Read Only	T or F: use F (Specifies whether the tag is editable in the database)
E Data Source	D, E, R, I, M, S (Device, DDE, Remote, I/O, Memory, System)
F Security Code	* (asterisk) or letters A-P: use * (A-P used by RSView only)
G Alarmed	T or F (not yet supported): use F
H Data Logged	T or F (not yet supported): use F
I Native Type	D, U, I, L, F, B, P, N, and 1-8 (Default, Unsigned Integer, Integer, Long, Floating Point, Byte, Bit Position, Binary, and 1BCD-8BCD)
J Value Type	L or F: use F if scale offset or initial analog value uses decimal digits, otherwise use L
K Min Analog	Minimum analog value: use 0
L Max Analog	Maximum analog value: use 100
M Initial Analog	Initial analog value (analog tag)
N Scale	Scale multiplier (val = Scale(X) + Offset) (analog tag)
O Offset	Offset (analog tag)
P Deadband	Deadband (unused): use 0
Q Units	Units label: use blank
R Off Label Digital	Digital off label: use blank
S On Label Digital	Digital on label: use blank
T Initial Digital	Initial digital value (must be either the on or off label): use 0 or 1 (digital tag)
U Length String	String length (string tag)
V Initial String	Initial string value (string tag)
W Node Name	Node name (device tag)
X Address	Programmable logic controller address (device tag)

Column	Description
Y Scan class	Scan class name: use A–K or custom names (device tag)
Z DDE Application	DDE application name: use blank
AA DDE Topic	DDE topic name: use blank
AB DDE Item	DDE item name: use blank
AC System Source	System data source name: use blank
AD System Source Index	System data source index: use blank
AE RIO Address	Remote I/O address (Remote I/O tags)
AF Element Size Block	Use B (block tag)
AG Number Elements Block	Number of bits in the block tag
AH Initial Block	Initial value of a block tag (0 or 1)

Creating a New Tag Database

The New Database command on the Database menu creates a new, empty tag database for the application. When you use this, all the tags in the current database, whether it is private or shared, are deleted. If the application has a shared database that no longer exists, a new shared database is created.

Creating Expressions

This chapter describes:

- types of expressions
- expression syntax
- using the Expression Editor
- adding, copying, editing, and deleting expressions
- assigning write expressions
- validating expressions

What Are Expressions?

Expressions perform mathematical or logical calculations to manipulate the display of tag values in the PanelView terminal. Expressions can reduce the complexity of PLC ladder logic by allowing the application in the PanelView terminal to make calculations and perform indicator logic normally done in the PLC.

For example, a temperature probe used to monitor the temperature of an oven is connected to a 12-bit analog input card in the PLC, but you want the PanelView terminal to display the temperature in degrees Celsius. You can create an expression to convert the temperature at the PanelView terminal, instead of creating a ladder rung in the PLC to perform the calculation (see Example 3, later in this chapter, for more information).

Note: If you assign expressions that contain any component other than tag names and comments, the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Tip: Use the information on the Remote I/O Tags or Device Tags Worksheets when creating expressions for your application.

Types of Expressions

You can use an expression to manipulate the value of any analog or digital tag that reads output from the PLC. You cannot assign expressions to the following:

- the ASCII Display object (because it requires String tags)
- the Transfer Inhibit Control
- the Bit-triggered Alarm controls

An expression can also perform calculations on the values of multiple read tags. However, expressions can perform calculations on analog and digital tags only. Expressions cannot perform calculations on string or block tags.

If a control allows you to use an expression, the Assign Expression button is available; otherwise, it is dimmed. The use of expressions is always optional. If you don't want to use expressions, you can assign tags to the controls.

PanelBuilder also supports *write expressions*. A write expression allows you to manipulate the value of a tag before it is written to the PLC. PanelBuilder supports write expressions for these objects:

- Numeric Entry Keypad
- Numeric Input Cursor Point
- Numeric Keypad Enable Button

For these three objects only, you can assign an optional write expression that performs calculations on the operator's input before sending it to a write tag. "Assigning Write Expressions," later in this chapter, describes how to create write expressions for these objects.

Note: The Numeric Input Cursor Point's Raise and Lower keys are disabled if it is assigned a write expression.

How Expressions Work

PanelView evaluates an expression whenever the value of a tag that is referenced in the expression changes. Multiple references to the same tag will have the same value.

Tag values are updated in a DH+ or ControlNet application based on the tag's assigned scan classes, which will affect when PanelView evaluates its expressions.

PanelView evaluates the expression until it encounters an Exit statement or until it has evaluated the last statement of the expression. Evaluation also stops if an "Unreliable" expression result or tag value is encountered. After PanelView has finished evaluating the expression, if the result is "Reliable," PanelView returns the evaluation result. This result is displayed on screen.

If one of these conditions causes the result to be Unreliable, a minor fault message is displayed:

- division by 0
- a value in the expression is out of range of the expression's Long or Single data type
- the expression calculation results in an out of range value
- a tag value is out of range
- the tag's address does not exist

If the expression is assigned to a pen in the Trend object, and the PLC communications are lost, the expression's value will be unreliable but no minor fault message will appear.

A major fault message is displayed if the PanelView terminal runs out of memory while evaluating the expression.

Important: If an expression uses division and the denominator is a tag, then a minor fault message is displayed when the application starts up, under these conditions only:

- if the expression is assigned to an object on the Application Startup Screen.
- if the expression is assigned to any of the alarm or Global PLC I/O controls.

Expression Data Types

Expressions perform mathematical and logical calculations on numerical and Boolean values. Expressions can work with values that use these data types:

Data Type	Description	Range
Boolean	A 32-bit value	True=1, False=0
Byte	An unsigned, 8-bit value	0 to 255
Integer	A signed, 16-bit value	–32,768 to 32,767
Long	A signed, 32-bit value	–2,147,483,648 to 2,147,483,647
Single (float)	An IEEE 32-bit floating point value	-3.402823×10^{38} to $-1.401298 \times 10^{-37}$ for negative values zero 1.401298×10^{-37} to 3.402823×10^{38} for positive values
Variant	A special 32-bit value that can contain any one of the Boolean, Byte, Integer, Long, or Single data types. It adjusts its data type to accommodate the value assigned to it.	See above. Has full range of all the other data types.

Tag Data Types in Expressions

You can only use an analog or digital tag in an expression.

The tag's data type affects how the tag's value is represented in an expression. Digital tags are represented with the Long data type, and their value is unaffected. For analog tags, if the data type is Float the tag is represented with the Single data type. All other analog tag data types are represented as the Long data type.

Note: PanelView ensures expressions have floating point accuracy of at least 7 decimal digits.

Expression Language

The PanelBuilder expression language is modeled after the Microsoft Visual Basic™ programming language and uses many of the same conventions. This section describes the syntax to use for the various components of your expressions.

Line breaks are meaningful in the PanelBuilder expression language. A line break can indicate a particular construct, or the end of a statement. Follow the syntax rules described in this section to ensure that your expressions perform as expected.

Keywords, tag names, and local variable names are case insensitive. That means you can enter them as upper case, lower case, or as a mixture. When expressions are compiled on .pvd file creation, PanelBuilder converts all keywords, tag names, and local variables to upper case.

Types of Operations Expressions Can Perform

You can use expressions to perform these types of operations on tag values:

- arithmetic operations
- comparison operations
- construct operations
- logical operations

Order of Precedence

Arithmetic operations take precedence over comparison operations, and comparison operations take precedence over logical operations. Use parentheses to override the order of precedence. Operations within parentheses are performed before operations outside the parentheses. Operations within parentheses are performed according to the precedence rules.

The following list shows the order of precedence (in descending order) for all operators:

( )
– (negation)
* and / (floating point division)
\ (integer division)
Mod
+ and – (subtraction)
= and <> and < and > and <= and >=
Not
And
Or
Xor

See the section for each type of operation for more information. For example, see "Arithmetic Operations," later in this chapter, for information about the order of precedence of individual arithmetic operators.

Other Expression Components

You can also use these components in your expressions:

Component	Symbol	Description
Braces	{ }	Use braces to enclose tag names. Use of braces is optional, but required when tag names contain these ambiguous characters: backslash (\), period (.), hyphen (–), and underscore (_). If you do not use braces, the ambiguous characters will be treated as expression language syntax rather than as part of the tag name. For example, there are tags called PUMP and PUMP–1. PUMP–1 will be treated as the value of tag PUMP minus 1. {PUMP–1} will be treated as the value of tag PUMP–1.
Comments	' (apostrophe) or REM	Use comments to describe the expression. A comment can start at the beginning of the line, or you can place comments to the right of executable code. You cannot place comments to the left of executable code. You can use up to 100 comments in each expression. Comments are removed when the expression is compiled. We recommend that you begin each expression with a comment; the first line of the expression is displayed in the Control list. Multi–line comments must include the comment indicator on each line. See Examples 2 and 3, later in this chapter, for samples of how to use comments.
Constants	any number	Constants are simply numbers that you use in expressions.
Exit statements	Exit	Exit statements allow you to explicitly terminate the evaluation of the expression. PanelView uses the value it calculated immediately before reading the Exit statement. Use of Exit statements is entirely optional. The Exit statement may follow any line of executable code. The Exit statement must be written on its own line, although it may be followed by a comment.
Line continuations	_ (underscore)	Use the line continuation character when a line of code is too long to read easily in the Expression editor. When used, the line continuation character must be the last executable character on the line. There must be a space or tab between the previous executable code and the line continuation character. You can use the line continuation character on consecutive lines.
Parentheses	( )	Use parentheses to override operation precedence in your expressions, and to add clarity. See Examples 1 through 3, later in this chapter, for samples of how to use parentheses.

Component	Symbol	Description
Spaces, tabs, and blank lines		You can use spaces, tabs, and blank lines wherever you like, to make your expressions clear and readable. The only place where a space is not permitted is between the negation operator and its operand. In some cases spaces are required. For example, between the line continuation character and the previous executable code. If in doubt, see the section that describes the expression component you are using. Extra spaces, tabs, and blank lines are removed when the expression is compiled.
Tags	{ *Tagname* }	You can use any analog or digital read tag in your expressions. You must use the full tag name, including any tag folder names. You can use the same tag multiple times in an expression. However, you may then wish to assign the tag to a local variable. You can use up to 50 different tags in each expression. See also "Braces," above.
User entry character	?	You must use the user entry character in the optional write expressions, if you assign them. It serves as a placeholder for the value the operator enters. The user entry character will be replaced by a numeric value and is treated as a Long or Floating Point constant. You cannot use the user entry character in any other expressions. See "Assigning Write Expressions," later in this chapter, for more information.

Local Variables

Local variables allow you to simplify your expression text and make it more readable. For example, if you are repeatedly referencing a tag with a long name, such as Tag_For_Temperature_Probe, you can create a local variable called "analog_input" and assign the tag to it. You can also use a local variable when you want to use a calculated value multiple times in an expression. See Example 5 at the end of this chapter. You can use up to 30 local variables in each expression.

You must declare a local variable in the expression before you use it. The declaration includes the name of the variable and, optionally, the data type.

Use this syntax to declare local variables:

Dim *varname1* [As *datatype1*] [, *varname2* [As *datatype2*]]…

Alternatively, you can declare each variable separately on its own line, with its own Dim statement.

For example:

```
DIM temperature AS SINGLE,
input AS INTEGER,
probe AS LONG
```

Variable names are a maximum of 40 characters long. They must begin with a letter from A to Z, and can contain letters, numbers, and the underscore character (_). Variable names are case-insensitive, and cannot be reserved keywords. See Appendix F, *Reserved Keywords*, for a list of reserved keywords.

Note: Assigning a data type to the variable is optional. If you do not assign a specific data type the variable will be the Variant data type. You can assign these data types to variables: Boolean, Byte, Integer, Long, Single, and Variant. See “Expression Data Types” earlier in this chapter for more information about data types.

For example:

```
DIM variable1, variable2 AS INTEGER
```

declares *variable1* as a variable of type Variant; *variable2* is declared as type Integer.

You can assign a value to a local variable. The value can include any combination of keywords, variables, tags, constants, and operators that result in a number.

Use this syntax to assign a value to a local variable:

Variablename = Value

You can assign different values to the same variable at different places in the expression. PanelView always uses the most recent assignment, which supersedes earlier assignments.

Important: You also use the equal sign “=” to perform the equality operation. To avoid confusion between the two operations, always place parentheses “( )”around the operands when you wish to perform the equality operation on a variable. Otherwise, PanelView may treat the operation as an assignment to the variable.

See Examples 2 and 3, later in this chapter, for samples of how to declare and assign values to variables.

Evaluation of Local Variables

If the value assigned to a variable uses a different data type than the variable’s data type, the value may be outside the range of the local variable. If this occurs the result of the assignment is “Unreliable.” If the value is within the range of the local variable’s data type, the value is converted to the local variable’s data type.

This table lists the resulting value after the value is converted to the local variable's data type:

	Local Variable Data Type:			
Value Data Type	**Byte, Integer, Long**	**Single**	**Boolean**	**Variant**
Byte, Integer, Long	Value is Unreliable if outside local variable's range; otherwise, value is unchanged.	Value is Unreliable if outside local variable's range; otherwise, value is unchanged.	If value = 0, resulting value is False. If value <> 0, resulting value is True.	Value unchanged.
Single	Value is truncated to an integer. Value is Unreliable if outside local variable's range; otherwise, value is the integer value.	Value unchanged.	If value = 0, resulting value is False. If value <> 0, resulting value is True.	Value unchanged.
Boolean	Value is 0 if False, 1 if True.	Value is 0 if False, 1 if True.	Value unchanged.	Value unchanged.
Variant	See above; resulting value depends on which data type the value is evaluated as.	See above; resulting value depends on which data type the value is evaluated as.	See above; resulting value depends on which data type the value is evaluated as.	Value unchanged.

Expression Maximums

The number of components in each expression must not exceed these maximums:

Component	Maximum
Tags	50
Local variables	30
Comments	100

There are no limits on the other types of components. However, the size of an expression's text cannot exceed 2048 characters. The size of the compiled expression (in the .pvd file) cannot exceed 1 Kb. The "Percent Full" indicator in the Expression editor alerts you to when you are approaching the size limit for an expression.

Arithmetic Operations

This table describes the types of arithmetic operations you can use expressions to perform. Operators are listed in order of precedence, and exceptions are noted below.

Operator	Symbol	Syntax	Description of Result	Data Type of Result
Negation	–	*–Operand*		If the operand is data type Single, the result is Single. Otherwise, the result is Long.
Multiplication	*	*Operand1 * Operand2*		If either operand is data type Single, the result is Single. If neither operand is Single, the result is Long.
Division	/	*Operand1 / Operand2*	Floating point (contains decimal places)	Single
Integer division	\	*Operand1 \ Operand2*	Integer (no decimal places)	Long
Modulo	Mod	*Operand1* Mod *Operand2*	Remainder (of an integer divide)	Long
Addition	+	*Operand1 + Operand2*		If either operand is data type Single, the result is Single. If neither operand is Single, the result is Long.
Subtraction	–	*Operand1 – Operand2*		If either operand is data type Single, the result is Single. If neither operand is Single, the result is Long.

There is no space between the negation symbol and the operand, while there are spaces before and after all the other arithmetic symbols.

Note: Some arithmetic operations take precedence over others. Operations with the highest precedence are evaluated first. However, Addition and subtraction have equal precedence, and multiplication and floating point division have equal precedence (over integer division). Operations with equal precedence are evaluated in sequence from left to right. Operands can include any combination of variables, tags, constants, and operators that result in a number.

Example 1: Using Arithmetic Operations in Expressions

A temperature probe used to monitor the temperature of an oven is connected to a 12–bit analog input card in the PLC, but you want the PanelView terminal to display the temperature in degrees Celsius. The temperature probe has a range of 100°C to 500°C. A temperature of 100°C produces an analog input of 0, and a temperature of 500°C produces an analog input of 4,095. The mathematical relationship between the two temperature scales is linear.

Create an expression to convert the analog input temperature reading to Celsius. To display the temperature using a Numeric Data Display object at the PanelView terminal, assign this expression to the Displayed Value control:

```
( {Tag_For_Temperature_Probe} * 400 \ 4095 ) + 100
```

The expression uses integer division (" \ "). The result will be truncated to an integer (with no decimal places). For higher accuracy when converting the analog value into a temperature, use floating point division (" / "):

```
( {Tag_For_Temperature_Probe} * 400 / 4095 ) + 100
```

This expression will produce a result that contains decimal places. If you have configured the Numeric Data Display to use the Implicit or Fixed Position decimal point option, the appropriate number of decimal places will be displayed. However, floating point math is much slower than integer math. You must weigh the tradeoff between speed and accuracy when considering which type of division to use.

Comparison Operations

Comparison operators have equal precedence. They are evaluated in sequence from left to right. Operands can include any combination of variables, tags, constants, and operators that result in a number. All comparison operations produce a result of the data type Boolean: True if the comparison is true, and False if it is not. The result is considered reliable if both operands are reliable.

This table describes the types of comparison operations you can use expressions to perform.

Operator	Symbol	Syntax
Equality	=	(*Operand1* = *Operand2*)
Inequality	<>	*Operand1* <> *Operand2*
Less than	<	*Operand1* < *Operand2*
Greater than	>	*Operand1* > *Operand2*
Less than or equal to	<=	*Operand1* <= *Operand2*
Greater than or equal to	>=	*Operand1* >= *Operand2*

There are spaces before and after all the comparison operators.

Important: You also use the symbol “=” to assign variables. To avoid confusion between the two operations, always place parentheses “( )”around the operands when you wish to perform the equality operation on a variable. Otherwise, PanelView may treat the operation as an assignment to the variable.

See “Construct Operations,” next, for examples using comparison operators in expressions.

Construct Operations

Construct operations allow you to use conditional logic in your expressions.

This table describes the types of construct operations you can use expressions to perform.

Operator	Syntax	Description of Result

IF..THEN	If *Condition* Then
	Statement(s)
	EndIf
		If the value of the condition is non–zero (True), the statements are executed and evaluation of the construct ends.
		If the value of the condition is zero (False), evaluation of the construct ends and returns a value of zero.
		Note: Conditions and statements can include any combination of variables, tags, constants, and operators that result in a number.

IF..THEN..ELSE	If *Condition* Then
	Statement(s)
	Else
	ElseStatement(s)
	EndIf
		If the value of the condition is non–zero (True), the statements are executed and evaluation of the construct ends.
		If the value of the condition is zero (False), the else statements are executed and evaluation of the construct ends.
		Note: Conditions, statements and Else statements can include any combination of variables, tags, constants, and operators that result in a number.

SELECT CASE	SelectCase *Test* Case *Constant1* *Statement(s)1* Case *Constant2* *Statement(s)2* : Case *ConstantN* *Statement(s)N* [CaseElse *ElseStatement(s)*] EndSelect	If the value of the test matches any of the constants, the statements following the first match are executed and evaluation of the construct ends. Use of CaseElse is optional. If you use it, and the value of the test does not match any of the constants, the else statements are executed and evaluation of the construct ends. If you do not use it, and the value of the test does not match any of the constants, evaluation of the construct ends. **Note:** Tests, statements and Else statements can include any combination of variables, tags, constants, and operators that result in a number. When using a range of constants, separate them with "To", for example, 1 To 5.

Note: Once PanelView has finished evaluating the construct, the terminal evaluates the statements following the EndIf or EndSelect keyword, if any.

Example 2: Using If..Then..Else Constructs in Expressions

The analog input temperature probe from the previous example does not produce linear results. The probe has three linear sections, each with a different slope:

Analog Input	Temperature in °C
0	100
1000	200
4000	400
4095	500

Use the If..Then..Else construct to create a new expression to convert the analog input temperature reading to Celsius. This example also uses variables and comments, described later in this chapter:

```
' This expression converts the analog input temperature
' probe value to degrees Celsius.

' The temperature probe produces the following output:
'             Analog       Celsius
'             0            100.0
'             1000         200.0
'             4000         400.0
'             4095         500.0

DIM analog_input AS INTEGER

' analog_input is the variable name for the analog input
' from the probe.

DIM temperature AS SINGLE

' Temperature is the variable name for the temperature
' in degrees Celsius. The Single data type yields a
' floating point decimal result.

analog_input = {Tag_For_Temperature_Probe}

' The analog_input variable is assigned to the tag that
' reads the analog input.

' The conditional expression wraps to the next line.
' Note the use of the line continuation character ( _ ).
```

```
IF ( (analog_input >= 0) AND (analog_input <= 1000) ) _
THEN
             temperature = ( analog_input * 100 / _
             1000 ) + 100

' Use this calculation if analog value is between
' 0 & 1000.

ELSE
             IF ((analog_input>1000) AND
(analog_input<=4000)) THEN
             temperature = ((analog_input – 1000) * 200
/ 3000 ) + 200

' Use this calculation if analog value is betw. 1000 &
' 4000.


ELSE
             temperature = ( (analog_input – 4000)
             * 100/ 95 ) + 400

' Use this calculation if analog value is betw. 4000 &
' 4095.

             ENDIF

ENDIF
```

Note that the expression uses nested If..Then..Else constructs. Nested constructs can be hard to read. Nested If..Then..Else constructs can often be replaced by a Select Case construct.

Example 3: Using the Select Case Construct in Expressions

This example restructures the expression in the previous example using the Select Case construct:

```
' This expression converts the analog input temperature
' probe value to degrees Celsius.

' The temperature probe produces the following output:
'            Analog       Celsius
'            0            100.0
'            1000         200.0
'            4000         400.0
'            4095         500.0

DIM analog_input AS INTEGER

' analog_input is the variable name for the analog input
' from the probe.

DIM temperature AS SINGLE

' Temperature is the variable name for the temperature in
' degrees Celsius. The Single data type yields a floating
' point decimal result.

analog_input = {Tag_For_Temperature_Probe}

' The analog_input variable is assigned to the tag that
' reads the analog input.

' The conditional expression wraps to the next line.
' Note the use of the line continuation character ( _ ).

SELECTCASE analog_input

             CASE 0 TO 1000
             temperature = ( analog_input * 100 / 1000 ) _
                         + 100

' Use this calculation if analog value is between 0
' and 1000.

             CASE 1001 TO 4000
             temperature = ((analog_input – 1000)
                         * 200 / 3000) + 200

' Use this calculation if analog value is betw. 1000
' & 4000.
```

```
            CASE 4001 TO 4095
            temperature = ( (analog_input – 4000)
                        * 100 / 95 ) + 400

' Use this calculation if analog value is betw. 4000
' & 4095.

            CASEELSE
            temperature = –1

' The probe should not be returning values outside the
' range of 0 to 4095. If the PanelView terminal displays
' –1 then the operator will know something is wrong.

ENDSELECT
```

Logical Operations

Some logical operations take precedence over others. Operations with the highest precedence are evaluated first. Operands can include any combination of keywords, variables, tags, constants, and operators that result in a number or Boolean value.

The logical operation differs depending on the data type of the operand:

Data Type of Operand:	Logic Operation Type:
If both are Byte, Integer, Long, Variant, or any combination of these	Bit-wise
If one or both are Boolean or Single	Boolean (1 = True, 0 = False)

In either case, the result is considered reliable if both operands are reliable.

This table describes the types of logical operations you can use expressions to perform. Operators are listed in order of precedence.

Operator	Data Type of Operands	Description	Syntax	If Operand1 Is:	And Operand2 Is:	Then Result Is:
Not	Byte, Integer, Long	Performs a bit-wise negation on the operand	Not *Operand1*	0 1		–1 –2
	Boolean or Single	Performs a logical negation on the operand		True False		False True
And	Byte, Integer, Long	Performs a bit-wise conjunction on the operands	*Operand1* And *Operand2*	0 0 1 1	0 1 0 1	0 0 0 1
	Boolean or Single	Performs a logical conjunction on the operands		True True False False	True False True False	True False False False
Or	Byte, Integer, Long	Performs a bit-wise disjunction on the operands	*Operand1* Or *Operand2*	0 0 1 1	0 1 0 1	0 1 1 1
	Boolean or Single	Performs a logical disjunction on the operands		True True False False	True False True False	True True True False
Xor	Byte, Integer, Long	Performs a bit-wise exclusion on the operands	*Operand1* Xor *Operand2*	0 0 1 1	0 1 0 1	0 1 1 0
	Boolean or Single	Performs a logical exclusion on the operands		True True False False	True False True False	False True True False

Example 4: Using Logical Operations in Expressions

With expressions, you can pack a large amount of information into a single word of data. The bit-wise operations AND, OR, and NOT can be used to extract information out of the packed data and used to set the state of a multi-state indicator, for example.

Consider a Symbol object representing a motor that controls a conveyor belt. The Symbol representing the motor on the PanelView terminal has four colors, each representing a possible state:

State Value	Color
0	Gray—not running, no alarms present
1	Green—running

State Value	Color
2	Bright Green—starting (run contact on and low speed contact off)
3	Red—not running; in alarm

The I/O for each starter has been mapped into file N7:0.

Bit Position	Description
/0	Overload Alarm
/1	High Temp/Thermistor
/2	Run Contact from Starter
/3	Low Speed Contact
/4	Phase Loss Alarm
/5	Under/Over Voltage Alarm
/6	Low Speed Alarm

The following expression uses tag motor1_sts, defined as N7:0, and sets the appropriate state at the PanelView terminal:

```
DIM state AS INTEGER       'current state of the motor
DIM result AS INTEGER      'result (decode status info)

REM All bits 0 means not running
REM Only bit 2 (value = 4) means running
REM Bit 2 (running) & Bit 3 (low speed) means starting
REM Any other bits means alarm state

state = {motor1_sts}       'Get current state of motor

IF ( state = 0 ) THEN
             result = 0    'The motor is not running
             EXIT
END IF

' The conditional expression wraps to the next line.
' Note the use of the line continuation character ( _ ).


IF (((state AND 4) <> 0) AND NOT ((state AND 251) _
     <> 0)) THEN
             result = 1    'The motor is running
             EXIT
END IF
```

```
IF (((state AND 4) <> 0) AND ((state AND 8) <> 0)  _
     AND NOT ((state AND 243) <> 0)) THEN
             result = 2    'The motor is starting, since
                           'run and low speed are active
             EXIT
END IF

IF ((state AND 243) <> 0) THEN
             result = 3    'The motor has an alarm
             EXIT

END IF
```

The numbers used are based on converting the binary representation of the state bits into a decimal number. For instance, when bit 2 is set, this is equivalent to the number 4; when bit 3 is set, this is equivalent to the number 8; 243 represents all alarm bits set.

Expression Return Values

It is important to understand how an expression is evaluated in order to correctly determine the result of the expression.

An expression will return the value of its last-executed statement.

In each of the following examples, *var* is a properly-declared local variable:

The expression:	Returns:
3	3
5 * 2	10
var	*var*
var = 3	3
(*var* = 3)	1 (True) if *var* is equal to 3; 0 (False) otherwise

IF (*var* = 3) THEN
5
END IF
	5 if *var* is equal to 3; 0 (False) otherwise

Expression Compilation and Regeneration

When you save the application as a .pvd file, and when PanelBuilder creates a .pvd file to download the application to the PanelView terminal, expressions are compiled into executable code. The executable code contains only the elements needed for evaluating the expression. These changes are made to the expression:

- comments are removed
- extra spaces, tabs, and blank lines are removed
- keywords and tag names are changed to upper case
- line continuation characters are removed and the connected lines are combined into a single line
- local variables are renamed as VAR0001, VAR0002, and so on. Where a conflict arises between a variable name and a tag name, the variable is assigned the next higher number.

The maximum size for the compiled expression is 1 Kb.

Note: An expression cannot be compiled if it contains syntax errors or invalid tag references. The .pvd file cannot be created until all errors are corrected. See “Validating Expressions,” later in this chapter, for more information.

When you open a .pvd file, PanelBuilder automatically regenerates the source text of expressions in the application. The expressions are regenerated so that they can be successfully compiled again. The executable code is identical to the original source text, with the following exceptions:

- the changes made during compilation remain in effect (see the list of changes above)
- all tag names that contain ambiguous characters are surrounded by braces { }
- expressions contain parentheses ( ) to indicate precedence order
- local variables that were not declared with a data type are now declared “AS VARIANT”

Using the Expression Editor

The Expression editor contains a text entry area where you type the expression. It also contains a list of components that you can use to create the expression. You cannot open the Expression editor until you have created an object to which an expression can be assigned. See *Types of Expressions*, earlier in this chapter, for more information about which objects can be assigned expressions.

To open the Expression editor:

1. Choose Configure Screen Object from the Edit menu, or double-click the object.
2. Select one of the object's controls, and choose Assign Exprn.

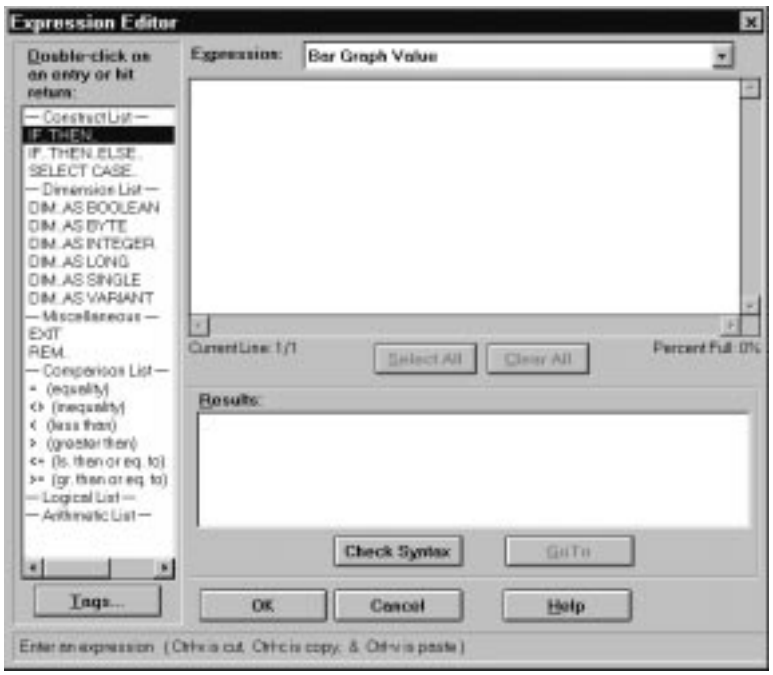
Note: The Assign Exprn button is enabled for controls which read values from the PLC. It is disabled for controls which write values to the PLC, with the exception of the following three objects, which support an optional write expression:

- Numeric Entry Keypad
- Numeric Input Cursor Point
- Numeric Keypad Enable Button

About the Expression Editor Window

The left-hand side of the Expression Editor window shows the list of components which can be used to construct an expression. You can double-click any of the headings to collapse or expand the list. The list is categorized as follows:

- Construct List: Constructs which are used to create an expression. Constructs are used to create a set of alternative courses of action when a particular value is found in the object.
- Dimension List: Data types which are used when you create variables for use in expressions. If you want to create a variable for use in an expression, you must declare its name, but declaring a data type for the variable is optional.
- Miscellaneous: Commands which may be used, each on their own line, anywhere in an expression.
- Comparison List: Comparison operators which are used for comparing values.
- Logical List: Logical operators which are used to include or exclude values, based on Boolean logic.
- Arithmetic List: Operators for performing arithmetic.



The right-hand side of the Expression Editor window is divided into two parts. The Expression section consists of a drop-down list containing all the controls for the dialog from which the Expression editor was launched. Below the control name is a text area where you enter and edit the expression.

The Results section consists of a message area where the results of Check Syntax are displayed. The Go To button is active only when a syntax error is displayed, and is used to go to the line in the expression at which the syntax error is found.

At the bottom of the Expression editor dialog is a status line, which displays messages about any selections you make.

Entering and Formatting Expressions

The Expression editor is a free-form text edit area, where you can enter expressions. The number of the current line is displayed beneath the text area, with a percentage indicator which displays the available memory space for the current expression. You can create expressions of up to 2,048 characters, with a maximum of 1,024 characters on a single line.

To select an expression component from the list, double-click an entry in the list. If the component requires arguments, the cursor flashes at the position where the first of the arguments is required.

You can indent lines of text by preceding the first character on the line with any number of spaces. When you press Enter to start a new line, the cursor is automatically moved to the position at which the first character of the previous line began. To revert to the margin, press the Backspace key until the cursor is at the left-hand margin, or, to move to a previous level of indent, press the Backspace key until the cursor is at the position of the previous level of indent. All subsequent lines will be indented at this position until you next change the indent position.

You can copy, cut, and paste text in expressions. First, highlight the text you want to copy or cut, or position the cursor at the position you want to paste text, then:

To do this:	Press this:
Copy	Ctrl + C
Cut	Ctrl + X
Paste	Ctrl + V

This also works for copying, cutting and pasting text between the Expression Editor and other Windows applications.

Checking Syntax

You can check the syntax of an expression at any time, by choosing Check Syntax.

If there are no errors in the syntax of the expression, the message "No syntax errors found in expression" is displayed in the Results message box. If there are syntax errors, the message "A total of *x* syntax errors were found in the expression," is displayed where *x* is the number of errors found. The errors, together with the line number on which they were found, are displayed beneath the first message in the Results message box.

To locate and correct a syntax error, double-click the message of the error you want to correct, or select the error message and choose Go To.

Saving Expressions

You can save all expressions for the current object or function.

To save changes to expressions:

- To save the expressions, choose OK; choose Cancel to discard all changes.

Assigning Write Expressions

Write expressions allow the PanelView terminal to perform a mathematical calculation or logical operation on the value the operator enters at the PanelView terminal. The result produced by the write expression is then sent to the Keypad Numeric Input control. The use of write expressions is optional.

Note: If the write expression attempts to divide by zero a value is not sent to the Keypad Numeric Input control.

Note: If you assign a write expression, the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

You can assign write expressions to three PanelBuilder screen objects:

- Numeric Keypad Enable Button
- Numeric Input Cursor Point
- Numeric Entry Keypad

Each of these objects has an Optional Keypad Write Expression control. If you assign the Optional Keypad Write Expression you must also assign a tag to the Keypad Numeric Input control, or PanelBuilder reports a validation error.

Note: If you assign an Optional Keypad Write Expression the Raise and Lower keys are disabled.

User Entry Character

The User Entry Character provides a placeholder in the expression for the value the operator enters in the terminal. The User Entry Character is a question mark “?”. You must enter at least one “?” in the write expression. You can enter multiple User Entry Characters in a single write expression.

Example 5: Entering User Entry Characters in Expressions

In this example, the user can regulate the speed of a conveyor belt, by entering a value using, for instance, a Numeric Entry Keypad. A latched push button defines the value as feet or meters per second. One state of the latched push button corresponds to feet per second, the other to meters per second. The tag is feet_or_meters and the PLC address is N7:0. If the user enters the value in feet per second, the value is passed to the PLC without conversion. If the user enters the value in meters per second, the write expression converts the value to feet per second before passing it to the PLC.

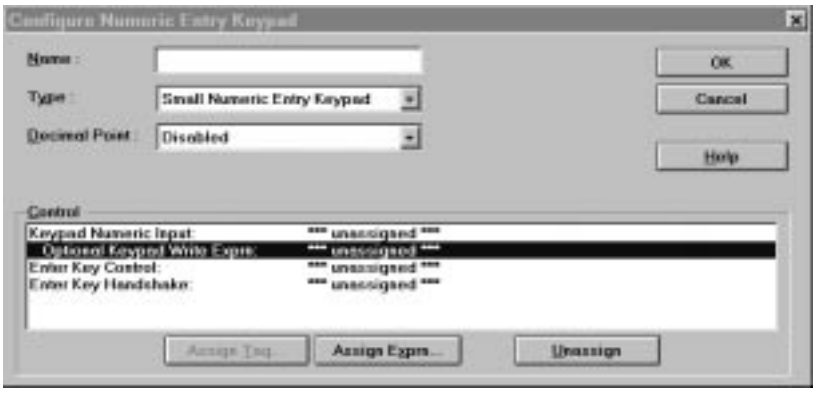
```
' If input_speed is entered in m/sec, convert to ft/sec
DIM input_speed AS INTEGER
input_speed=?

' If feet_or_meters = 1, convert input_speed from ft/sec
' to m/sec
IF feet_or_meters=0 THEN
input_speed
ELSE
input_speed*3.281
END IF
```

The “?” character is the placeholder for the value the operator enters at the terminal.

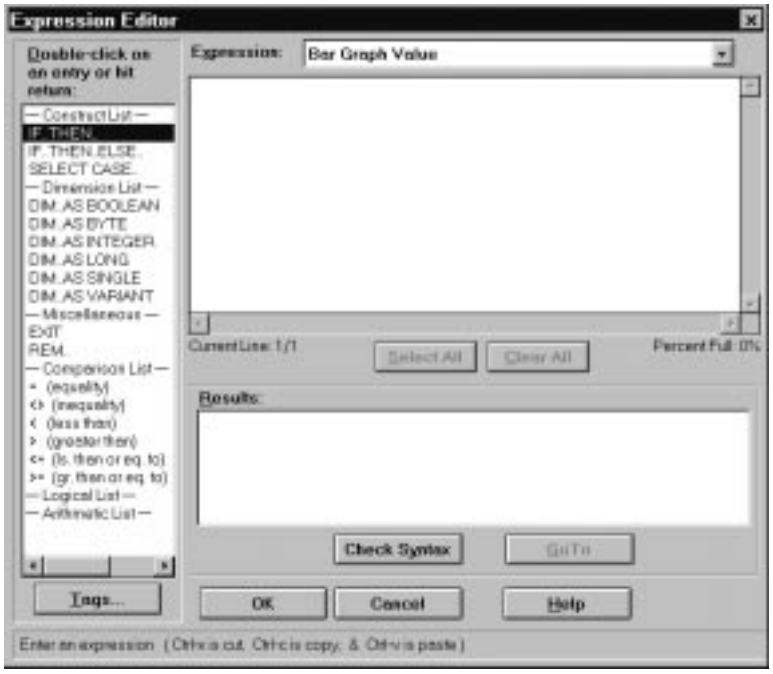
To assign a write expression:

1. In the object’s configuration dialog box (in our example, the Numeric Entry Keypad’s configuration dialog box for a touch screen application or the Numeric Keypad Enable Button’s configuration dialog box for a keypad application), select the Optional Keypad Write Exprn control.



2. Choose Assign Exprn.

   The Expression editor appears.



3. Enter the Expression in the Expression editor. Use the User Entry Character, ?, to indicate the position of the value entered by the operator.

4. Choose OK.

5. Assign the Keypad Numeric Input write tag. The numeric result of the write expression will be written to this tag.

Validating Expressions

You can perform a syntax check on your expressions as you create them, as described in “Using the Expression Editor,” earlier in this chapter.

Expressions are also validated when you perform a screen validation or application validation. Both these types of validation check expression syntax and verify that all tag references are valid.

If expression syntax errors in screen objects are discovered, PanelBuilder reports the screen, object, and expression that contains the error. For expression syntax errors in global functions, PanelBuilder identifies the function and expression that contains the error. Tag error messages identify the tag name, nature of the error, expression, and line of the expression containing the error.

A validation error also occurs if an optional write expression is assigned but the associated numeric input write tag is not assigned.

More Expression Examples

Use the following examples to help create your own expressions:

Example 6: Arithmetic Operations Provide Numeric Display Calculations within PanelView

Numeric Display Object:

56 % GOOD PART YIELD

Expression:

```
(GOODPART_TAG / TOTALPART_TAG) * 100
```

Example 7: Logical Operators Provide Indicator Logic within PanelView

Multistate Indicator Object:

Expression:

```
(MTR1_tag AND NOT MTR1_OL_tag)
   OR (MTR2_tag AND NOT MTR2_OL_tag)
   AND PRESS_SW_tag
```

Example 8: Comparative Operators Allow Threshold Logic within PanelView

Multistate Indicator Object:

State 0	State 1	State 2
LOW	MED	HIGH

Expression:

```
IF TEMP_tag < 100 THEN
0
ELSE IF (TEMP_tag >= 100) AND (TEMP_tag < 200) THEN
1
ELSE IF (TEMP_tag >= 200) THEN
2
END IF
```

Example 9: Numeric Write Expressions Allow Input to PLC

Numeric Input Object: Maintained PanelBuilder Object:

32 deg F ➡ 0 deg C ➡ PLC

State 0	State 1
DEG C	DEG F

Numeric Write Expression:

```
IF PB_tag = 0 THEN
?
ELSE IF PB_tag = 1 THEN
5 / 9 * (? – 32)
END IF
```

Example 10: Local Variables Improve Performance on Complex Expressions

Expression:

```
DIM Var1 AS INTEGER
Var 1 = ((PRESS_tag * 3 * TEMP_tag) / 5) + 6

IF (Var1 >= 3095) AND (Var1 < 6000) THEN
1
ELSE IF (Var1 >= 6000) THEN
2
END IF
```

Creating Screens

Application screens are the screens that appear on the PanelView terminal when the application is running. You create and design them in the Screens editor.

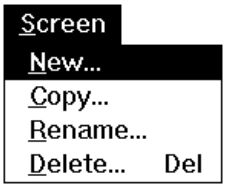
This chapter shows you how to:

- create new screens
- open, copy, and rename screens
- update and close screens
- delete screens
- set the background color
- assign screen security
- validate screens
- view the tags used in a screen
- use grids
- zoom out to display function keys
- assign the Application Startup screen

Creating a New Application Screen

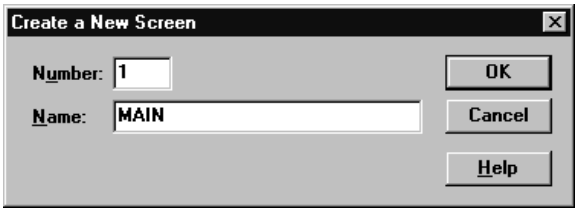
To create a new screen for your application, your Application Window must be open and active.

To create a new screen:



1. Choose New in the Screen menu or click on the toolbar. If the Screens folder is empty, you can double-click it to create a new screen.

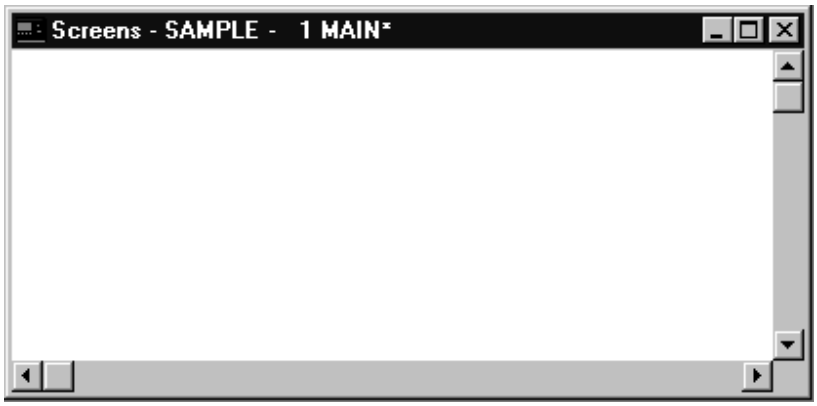
   The Create a New Screen dialog box appears. The next available screen number appears in the Number field.



2. To change the screen number, type the number you want to assign in the Number field. The number must be between 1 and 255, and must be unique. The number 255 is the default number for the Alarm History screen. If you assign this number to another screen, an error message appears. You must either assign a new number to the Alarm History screen or assign a different number to the application screen.
3. In the Name field, type the name for your screen, up to 15 characters.
4. Choose OK.

If this is the first screen you created for this application, a message appears, informing you that the Application Startup screen has been reassigned to this screen. You can change this if you wish. See "Assigning the Application Startup Screen," later in this chapter, for information.

The new screen appears. It shows the name and number you assigned.



Note: An asterisk (*) in the screen's title bar indicates that changes made to the screen haven't yet been saved.

Opening Screens

You open existing screens through the Application Window.

To open a screen:

To create a new screen for your application, your Application Window must be open and active.

1. Open the Screens folder.

   The list of available screens appears.

2. Choose the desired screen.

   The screen is displayed in a separate window on your workspace.

You can have multiple screens open at a time. For more information on working with multiple screens, see Chapter 3, *Creating Multistate Indicators and Drawing Lines*, in the *Getting Started* manual.

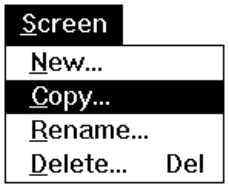
Copying Application Screens within the Same Application

You can copy an existing application screen, and make it into a new screen within the same application. This is an efficient way to create several similar screens quickly.

Tip: An easy way to ensure consistency between screens is to create a screen and make it into a screen template. Copy this template for each screen in the application, and add to it as required.

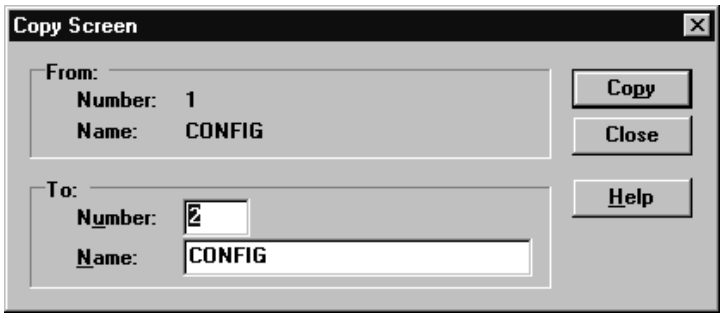
To create a new screen by copying an existing screen:

1. Open the Screens folder and select the screen you want to copy.



2. Choose Copy from the Application Window's Screen menu.

   The Copy Screen dialog box appears, with the name and number of the screen you are copying.



3. Type a new screen number in the Number field.
4. Type a new screen name in the Name field.
5. Choose Copy.

This copies the screen, its objects, and all their tag assignments to the new screen. If you wish, you can continue copying this screen. Copy as many screens as you wish.

6. When you're ready to exit, choose Close.

Important: Since the object controls (tag assignments and expression definitions) in the application screen are also copied, you may need to change them.

Copying Application Screen Contents into Another Application

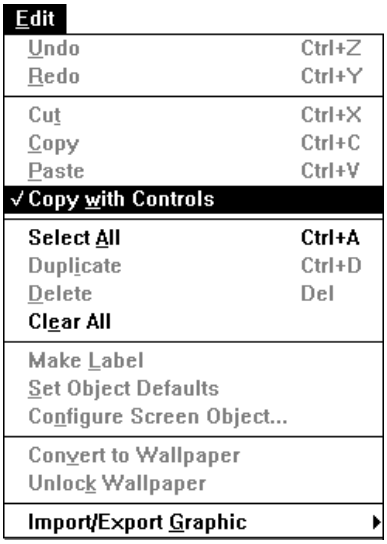
You can do a "screen copy" within the *same* application but not across multiple applications. However, you can copy certain objects from a screen in one application, and paste them into a screen in another application.

Note: Although you can copy some objects between touch screen and keypad applications, you can't copy objects that are specific to touch cell or keypad applications, or objects that contain buttons. This includes objects such as Control List Selectors and Push Buttons.

By default, when you copy objects, their controls (tag assignments and expression definitions) are also copied. You can choose whether or not to copy the controls.

Note: Even if you copy the control tag assignments and expression definitions, the tags themselves are not added to the destination application's tag database. You must add the tags to the destination application's tag database before the application will validate successfully. You may need to add the supporting racks and block transfer files (for Remote I/O tags) or nodes and scan classes (for DH+ or ControlNet tags).

Tip: If copying objects across applications using the same tags, make the tag database Shared and have both applications use it. Or, if you want to keep the tag databases Private, export the source application's tags and import them into the destination application.



To choose whether or not to copy controls:

- Choose Copy with Controls in the Edit menu. If the Copy with Controls menu item has a check mark (✓) beside it, the option is selected. If there is no check mark, the option is deselected.

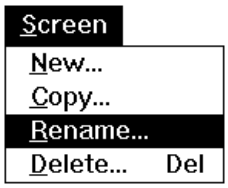
To copy a screen's contents into the screen of another application:

1. Open up both applications.
2. Open the screen you want to copy.
3. Select the objects you want to copy. If you want to copy the entire screen contents, choose Select All from the Edit menu, or press CTRL+A.
4. Choose Copy from the Edit menu or from the toolbar, or press CTRL+C.
5. Create a new application screen in the application you want to copy the screen to.
6. Click anywhere on the new screen window or within the screen.
7. Choose Paste from the Edit menu or from the toolbar, or press CTRL+V.

   The objects copied from the original screen are copied into the new screen, in exactly the same location as on the original screen.
8. If you copied the objects with their tag assignments, (that is, if Copy with Controls was checked in the Edit menu), you must add the tags' definitions to the database if the tags don't already exist.

Renaming Screens

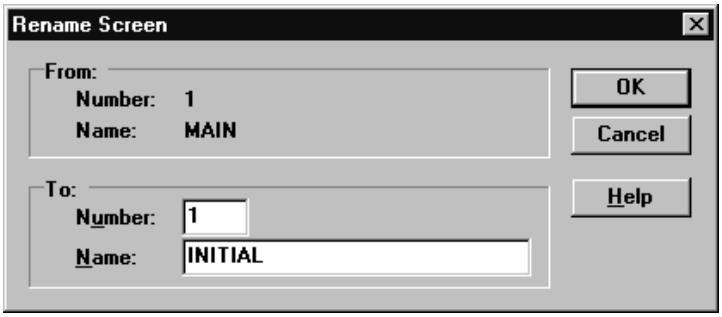
You can change the name and number of an existing screen with the Rename command.



To rename a screen:

1. Select the screen you want to rename in the Screens folder.
2. Choose Rename from the Application Window's Screen menu or from the toolbar.

The Rename Screen dialog box opens with the current number and name.



3. Select the Number field and type in the new number.

Note: The number 255 is the default number for the Alarm History screen. If you assign this number to a screen, an error message appears. You must either assign a new number to the alarm screen or assign a different number to the application screen.

4. Select the Name field and type in the new name. The name can be up to 15 characters long.
5. Choose OK.

   The new screen name and number replace the original name in the Screens editor.

Updating Screens

When you close a screen, the changes you made to your screen are updated automatically.

Important: Updating your screen saves the screen only temporarily. To save your changes permanently you must also save the application.

Closing Screens

When you have finished working in a screen, you can close it.

To close a screen:

- Choose Close from the Screen menu, double-click the screen's Control menu box, or press CTRL+F4.

  Any changes you made are updated automatically.

Deleting Screens

You can remove application screens from an application.

Note: If you delete a screen that contains objects with defined initial state values, the tag's initial value will be used. However, if an object with the defined initial state value in another screen has the same tag assigned, its initial state value will be used.

To delete a screen:

1. Open the Screens folder in the Application Window.
2. Select the screen you want to delete.
3. Choose Delete from Screen menu, press the DELETE key on your keyboard, or from the toolbar.

   The Delete dialog box asks you to confirm that you want to delete the screen.
4. Choose OK.

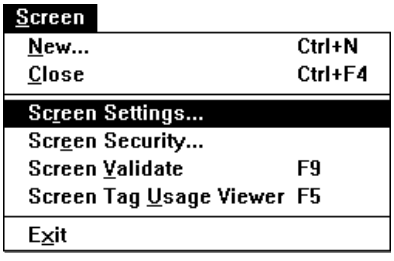
   The screen is deleted from the application.

Setting a New Screen Background Color

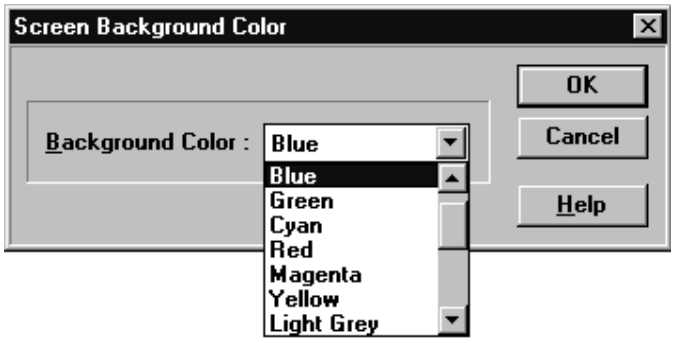
You can change the background color of the screen. The default background color is black.

To change a screen's background color:

1. Open the Screens folder, and open the desired screen.
2. Choose Screen Settings from the Screen menu.

   The Screen Background Color dialog box appears.
3. Select the color you want for your screen background from the Background Color drop-down list.



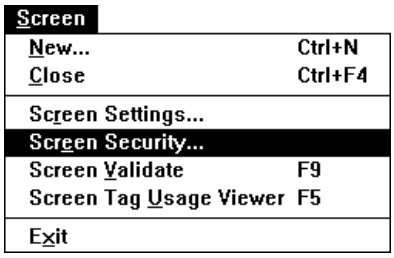
4. Choose OK.

   The screen's background color changes to the color you selected. All new screens you create will have the new color selected automatically.

Assigning Security Access to Screens

You may want to prevent some operators from accessing certain application screens. You can control operators' access by assigning a security classification to these screens. Up to sixteen operators can be given access to each screen.

Important: You can't assign security to the Alarm History or Alarm Status screens, since these must be available to every operator. Although you can assign security to the Application Startup screen and to screens requested by the programmable controller, any operator will be able to view these screens.



To assign security access to a screen:

1. Open the screen to which you want to assign security.
2. Choose Screen Security from the Screen menu.

   The Secure Screen dialog box appears.

3. For each operator you want to assign access to, either click the check box, or use TAB or the arrow keys to move through the selections and press SPACEBAR to make the selection.
4. When you've finished assigning security, choose OK.

To view this screen when the application is running, the operator must enter a 5-digit access code on a Security Entry window.

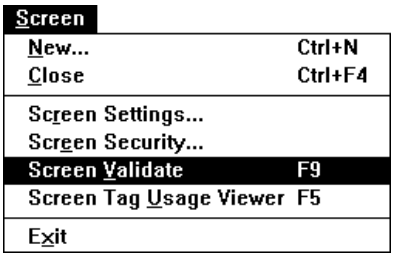
Note: If you don't specify operator authorization, the terminal won't request an access code, and anyone can access this screen. If you authorize one or more operators, the terminal will request an access code before it displays the screen.

Access codes are assigned on the terminal. A plant floor supervisor with a mode select key can view and change access codes without changing the application. For information on assigning and changing access codes on the terminal, refer to the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual,* Chapter 5, *Configuring PanelView Operator Terminals.*

Validating Screens

Before closing your application screen, you should ensure that there are no errors, such as overlapped input objects or objects that are off-screen.

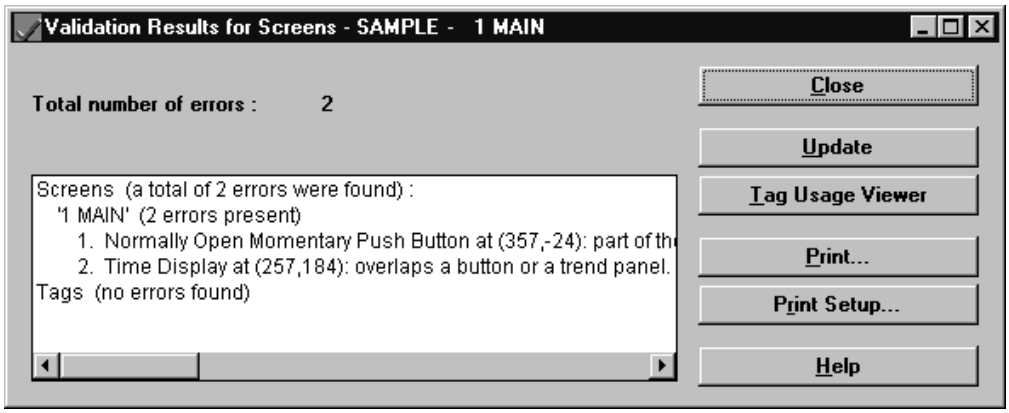
Note: Although you can close and save screens that contain validation errors, you can't download the application file if the screens contain errors. If you try to download with errors present, the application's Validation Results window appears.



To validate a screen:

1. Choose Screen Validate from the Screen menu or press F9.

   PanelBuilder checks your screen for errors. When the validation is finished, a Validation Results window appears, listing any errors in the screen and providing the pixel coordinates for the object causing the error. If there are no errors, the Validation Results window has a check mark.


Validation Results for Screens - SAMPLE - 1 MAIN
Total number of errors : 2
Close
Update
Tag Usage Viewer
Print...
Print Setup...
Help
Screens (a total of 2 errors were found) :
'1 MAIN' (2 errors present)
1. Normally Open Momentary Push Button at (357,-24): part of th
2. Time Display at (257,184): overlaps a button or a trend panel.
Tags (no errors found)

You can print the error report by choosing Print in the dialog box, or you can leave this dialog box open while you're making corrections to your screen.

2. Make all necessary corrections to your screen.

   Consult "PanelBuilder Validation Errors" in Appendix D, *Troubleshooting*, to help correct errors in your application that were caught in the screen validation.

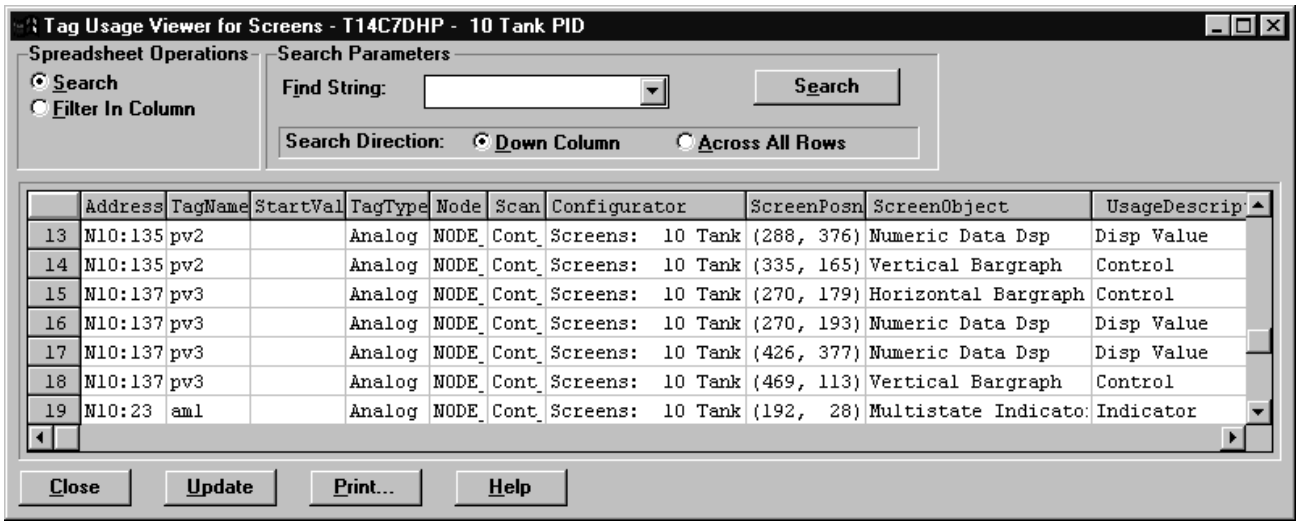
   Tip: With no objects selected, move the pointer over the screen. The coordinates of the pointer will be displayed in the status bar. This will enable you to find problem objects quickly.

3. When you've finished making changes, choose Update in the dialog box, or press F9, to check whether you've fixed the errors.

4. When you've fixed all errors, choose Close.

Viewing the Tags Used in a Screen

To see which tags have been assigned to the various objects in a screen, use the Tag Usage Viewer for Screens. The Viewer has search and filter functions so you can choose which tag information is displayed.


Tag Usage Viewer for Screens - T14C7DHP - 10 Tank PID
Spreadsheet Operations
Search
Filter In Column
Search Parameters
Find String:
Search
Search Direction: Down Column Across All Rows
Address TagName StartVal TagType Node Scan Configurator ScreenPosn ScreenObject UsageDescrip
13 N10:135 pv2 Analog NODE_ Cont_ Screens: 10 Tank (288, 376) Numeric Data Dsp Disp Value
14 N10:135 pv2 Analog NODE_ Cont_ Screens: 10 Tank (335, 165) Vertical Bargraph Control
15 N10:137 pv3 Analog NODE_ Cont_ Screens: 10 Tank (270, 179) Horizontal Bargraph Control
16 N10:137 pv3 Analog NODE_ Cont_ Screens: 10 Tank (270, 193) Numeric Data Dsp Disp Value
17 N10:137 pv3 Analog NODE_ Cont_ Screens: 10 Tank (426, 377) Numeric Data Dsp Disp Value
18 N10:137 pv3 Analog NODE_ Cont_ Screens: 10 Tank (469, 113) Vertical Bargraph Control
19 N10:23 aml Analog NODE_ Cont_ Screens: 10 Tank (192, 28) Multistate Indicato: Indicator
Close
Update
Print...
Help

To open the Tag Usage Viewer for Screens, do one of the following:

- choose Screen Tag Usage Viewer from the Screen menu
- press ALT+U
- press F5

Note: You can also view all the tags in an application using the Tag Usage Viewer for Applications. For details, see Chapter 6, *Defining Tags*. Chapter 6 also provides information on searching for strings, and filtering, updating, and printing the spreadsheet.

Working with Grids

These grids are available as guides for positioning, sizing and aligning objects:

- screen grid
- touch screen grid, if your application will run on a PanelView 1200 or 1200e terminal

Using the Screen Grid

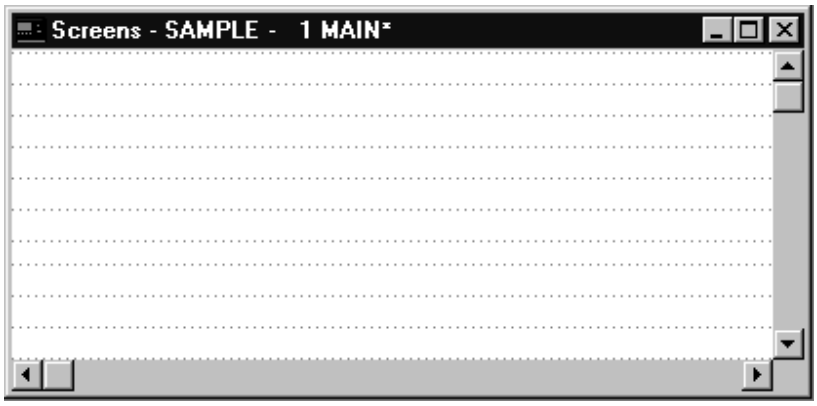
How you'll use the screen grid depends on the application type:

- For keypad applications, use the grid to align objects and design screen layout.
- For touch screen applications, use the grid to align text, graphic and display objects. For a PanelView 1200 or 1200e application, all input objects, such as push buttons, automatically align to the touch cells, even if the touch cell grid isn't selected or displayed.

  The PanelView 1000e and 1400e touch screen terminals use analog-resistive touch screens. This allows input objects to be any size (minimum 40 by 40 pixels) and to be placed anywhere on the screen. The analog-resistive touch screen provides greater flexibility in screen design compared to PanelView 1200 and 1200e terminals. PanelView 1200 and 1200e terminals use a touch matrix which requires that all input objects align with a touch grid.

ATTENTION: Do not press multiple touch screen locations at the same time. In the unlikely event that two or more touches occur simultaneously in the same horizontal or vertical line, an input object located half-way between the two touches could be activated. This is inherent in the analog-resistive touch screen technology used in PanelView 1000e and 1400e terminals.

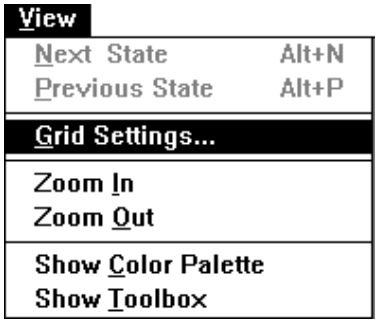
Every PanelView terminal screen is 640 by 480 pixels in size. You can change the grid spacing to enable you to size and position objects more precisely. The following illustration shows the default grid setting (8 by 20 pixels) for a keypad application screen.



If you don't want the grid to show, you can turn it off.

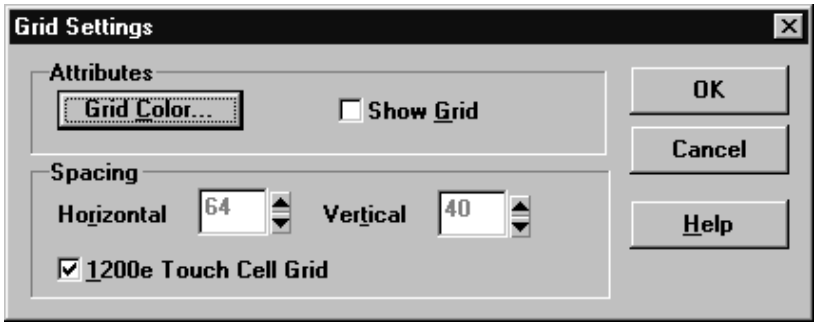
Note: If you change the grid settings, all screens that you open in the future will default to the new grid settings.

To change the default screen grid settings:



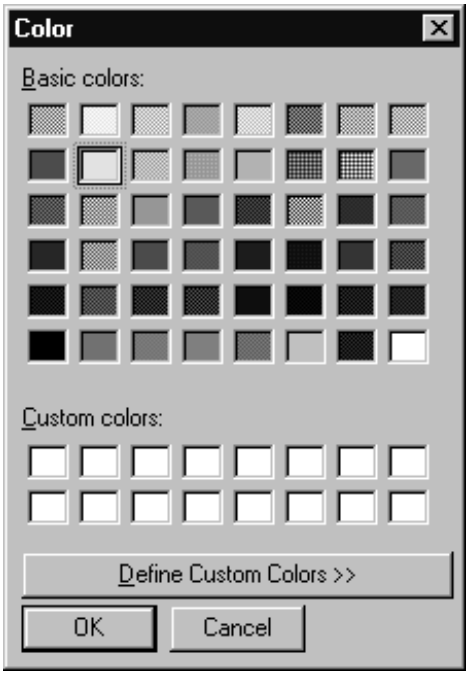
1. Choose Grid Settings from the View menu.

   The Grid Settings dialog box appears. You can choose not to show the grid, or you can change the grid sizes of the X- (horizontal) and Y- (vertical) axes. You can also change the grid color.



2. To hide the grid, clear the Show Grid box.
3. To hide the 1200e Touch Cell Grid, clear its box.
4. To increase or decrease the spacing of the grid lines on either axis, click the up or down arrow next to the appropriate field, or type a number in the box. For example, a value of 8 creates a grid line every 8 pixels.
5. To change the color of the grid, click on the Grid Color button.

The Color dialog box appears, showing the colors you can choose for your grid. You can select one of the Basic Colors, or you can choose the Define Custom Colors button and choose an area of the palette you want to use. It will appear in the Custom Colors section. You can then select it.



6. Select the color you want.
7. Choose OK.

Using the Touch Cell Grid (for PanelView 1200e Applications Only)

The touch cell grid indicates the touch cells that are available on the PanelView 1200e terminal. In 1200e touch screen applications, all input objects such as push buttons automatically align themselves to the touch cell grid.

To show the touch cell grid:

1. Choose Grid Settings in the View menu.

   The Grid Settings dialog box appears.
2. Select the 1200e Touch Cell Grid checkbox.

   You can't change the size of the touch cell grid.
3. Choose OK.

   The touch cell grid appears on the application screen.

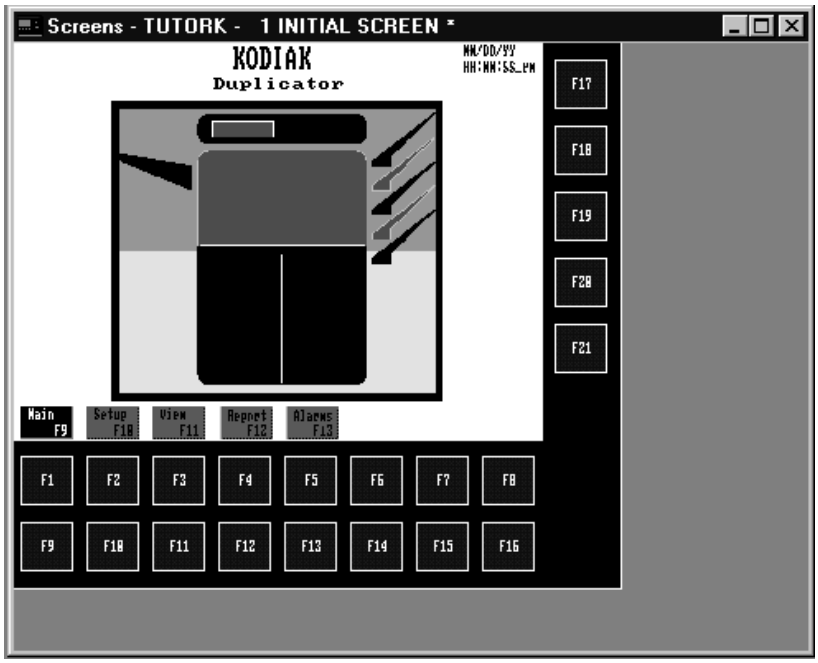
Displaying Function Keys (Keypad Applications Only)

When you're creating application screens for keypad terminals, you may want to align input objects to the function keys that are assigned to them. PanelBuilder enables you to display the function keys for your keypad application screens.

To display the Function keys:

- If you have Super VGA resolution, choose Maximize from the Control menu or in the screen's title bar.
- If you have VGA resolution, choose Zoom Out in the View menu.

  The screen appears with the function keys displayed. With Super VGA resolution, the screen appears as in the following illustration:



If your computer has Super VGA resolution, to return to normal view you can resize the screen or choose the button.

If your computer has VGA resolution, choose Zoom In from the View menu to restore the screen to normal size.

Assigning the Application Startup Screen

The Application Startup screen is the first screen that appears when you start the application.

In an application file without screens, the Application Startup screen is preset to the Alarm History screen (screen 255 by default), if the Alarm History screen is enabled. The first screen you create in an application is automatically assigned as the Application Startup screen. If you wish, you can assign another screen as the Application Startup screen.

If you delete the Application Startup screen, the screen with the next-highest number will automatically be assigned as the Application Startup screen.

Important: When the PanelView terminal is powered up, the Application Startup screen is always displayed, regardless of the security assigned to it.

To change the Application Startup screen:

1. In the Application Window, open the System folder and choose Terminal Setup, or choose from the toolbar.
2. In the General tab, click on the Screen field.
3. Choose the screen you want to assign from the drop-down list that appears.

Creating Objects

This chapter provides an overview of PanelBuilder 1400e objects. It tells you about:

- what objects are and do
- different types of objects
- working with objects
- creating objects
- adding labels to objects
- editing objects
- arranging objects
- configuring object states
- assigning tags and expressions to objects

This chapter does not provide details about individual application screen objects. For information on specific objects, see the *PanelBuilder 1400e Configuration Software for Windows Reference Manual*.

Tip: Before you begin creating objects and screens, use worksheets to plan your application. For details, see Chapter 2, *Planning Applications*.

What Is an Object?

Objects are the components that make up each application screen appearing on the terminal. Objects represent the components of traditional control panels, such as push buttons, indicators, or bar graphs. Objects also have different properties that allow them to perform the functions required for an automated process.

Objects allow the operator to monitor and control the operations of the process when the application is running.

Types of Objects

Some objects are available only for touch screen applications. Others are available only for keypad applications. Most are available for both. PanelBuilder automatically makes the appropriate objects available for your application type.

Important: If your application is configured for a touch screen terminal, you can't change it to run on a keypad terminal, or vice versa.

Dynamic and Static Objects

Objects are dynamic or static, depending on whether or not they interact with the programmable controller.

- **Dynamic objects** communicate with the programmable controller by receiving or transmitting information. You assign one or more tags or expressions to each dynamic object. You can use tags to write to or read from the programmable controller. In general, you can use expressions to read from the programmable controller only. For more information about tags see Chapter 6, *Defining Tags*. For more information about expressions see Chapter 7, *Creating Expressions*.
- **Static objects** appear on the screen, but don't interact with the programmable controller. Therefore you don't assign tags or expressions to these objects. Some examples of static objects are display text and lines.

Display and Input Objects

Dynamic objects are either display, input, or display and input objects, depending on how they interact with the programmable controller.

- **Display objects** receive information from the programmable controller. These objects usually display information about the status of the process. For example, the programmable controller places a value in the control tag of a bar graph. The bar graph object reads this value and displays it visually on the application screen. Display objects are also called *read objects*. They can use tags and expressions.

  All display objects are non-retentive.
- **Input objects** transmit information to the programmable controller. These objects usually control the application. For example, Increment and Decrement Value buttons could be configured to control the pressure in a storage tank. If the operator presses the Increment Value button, the new value is written to the button's tag at the programmable controller. The programmable controller uses this value to increase the pressure accordingly. Input objects are also called *write objects*. Input objects generally use tags only. All input objects are retentive.
- **Display and input objects** receive information from and transmit information to the programmable controller. For example, a Momentary Push Button sends information to the PLC to control a machine process, and the PLC sends information back to the push button to indicate the state the button should display. An object that both displays information and provides input to the PLC is also called a *read and write object*. It can use both tags and expressions.

Except for Momentary and Latched Push Buttons, all display and input objects are retentive.

About Retentive Objects

If an application is selected and run for the first time, or if power is switched off and on, a non-retentive object automatically reverts back to its initial values. Retentive objects can either revert to their Initial State values or, depending on the application type, do this:

RIO

- retain the values of their last active states

DH+ and ControlNet

- use the current programmable controller values

Note: Scrolling Lists can be configured to retain the values of their last active states for all application types.

The objects' runtime behavior depends on how you configure the Application Startup screen on the PanelView terminal. For more information on configuring this screen, refer to the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual.*

Note: If the terminal is switched from Run to Configure mode and back again or if the operator changes screens, retentive objects always retain their last values in Remote I/O applications. DH+ and ControlNet objects do not retain their last values, except for the Scrolling List object (if it has been configured to retain its last state).

Initial Tag Values and Initial State Values

All tags have initial values. If a tag is used as a write tag, its initial value is used to initialize the programmable controller when the application is first downloaded. It can also be used to initialize the programmable controller after a power cycle.

Certain objects have an initial state value. For some, it is predefined and cannot be changed. For others, you can optionally specify the initial state. When an initial state is defined for an object, the state value of this state overrides the object's control tag's initial value. The object's initial state value is used to initialize the programmable controller.

Note: If the initial state is not defined, the object state value will not override the control tag's value. Two objects with different initial state values cannot share the same tag, or a validation error will occur.

Safety Considerations for Retentive Objects

Instead of using a retentive object to start a machine or process, use a momentary input object such as a Momentary Push Button.

Also, make sure that the programmable controller is programmed to stop any machines or processes whenever the communication is interrupted, so that a machine or process can only be restarted by an operator.

ATTENTION: Never use a retentive object to initiate a process or turn on a machine. For example, imagine your programmable controller is programmed to turn a machine on whenever a Maintained Push Button is set to 1. If communication is temporarily cut while the Maintained Push Button is set to 1, the machine will automatically restart as soon as communication resumes.

Retentive Objects and Tags

Retentive objects reduce programmable controller programming instructions.

Every retentive object must have a tag that writes to the programmable controller.

More than one retentive object can be assigned to the same tag. For example, you can assign control selectors from different screens to the same tag. When one selector changes the tag value, other selectors will reflect the change.

Important: Momentary Push Buttons and Latched Push Buttons have a fixed initial state. This forces the buttons' control tags to the initial state value. Therefore, if other objects share the same control tag with these buttons, their initial state values cannot be different from those of these buttons.

ATTENTION: Make sure that the initial state values you define are the same values that the programmable controller program considers as the Off or default state.

The following table lists all the objects, and shows whether they are application-specific. It also lists some of their properties.

Object	Type	Keypad or Touch Screen	Read or Write	Retentive
Arc	Static	Both	Not applicable	N/A
ASCII Display	Dynamic	Both	Read	Non-retentive
ASCII Input (large or small)	Dynamic	Both	Both	Yes
Bar Graphs (vertical or horizontal)	Dynamic	Both	Read	Non-retentive
Control List Selector with Enter Key	Dynamic	Both	Both	Yes
Control List Selector without Enter Key	Dynamic	Both	Write	Yes
Date Display	Dynamic	Both	Not applicable	N/A
Decrement Value Button	Dynamic	Both	Write	Yes
Ellipse	Static	Both	Not applicable	N/A
Goto Configure Mode	Static	Both	Not applicable	N/A
Goto Screen Button	Static	Both	Not applicable	N/A
Graphic Image	Static	Both	Not applicable	N/A
Increment Value Button	Dynamic	Both	Write	Yes
Interlocked Push Button	Dynamic	Both	Write	Yes
Latched Push Button	Dynamic	Both	Both	Non-retentive
Line	Static	Both	Not applicable	N/A
List Indicator	Dynamic	Both	Read	Non-retentive
Local Message Display	Dynamic	Both	Read	Non-retentive
Maintained Push Button	Dynamic	Both	Both	Yes
Multistate Indicator	Dynamic	Both	Read	Non-retentive
Multistate Push Button	Dynamic	Both	Both	Yes
Normally Closed Momentary Push Button	Dynamic	Both	Both	Non-retentive
Normally Open Momentary Push Button	Dynamic	Both	Both	Non-retentive
Numeric Data Display	Dynamic	Both	Read	Non-retentive
Numeric Entry Keypad (small and large)	Dynamic	Touch Screen	Both	Yes
Numeric Input Cursor Point	Dynamic	Keypad	Both	Yes

Object	Type	Keypad or Touch Screen	Read or Write	Retentive
Numeric Keypad Enable Button	Dynamic	Keypad	Both	Yes
Panel	Static	Both	Not applicable	N/A
Rectangle	Static	Both	Not applicable	N/A
Return to Previous Screen Button	Static	Both	Not applicable	N/A
Scale	Static	Both	Not applicable	N/A
Screen Keypad Enable Button	Static	Keypad	Not applicable	N/A
Screen List Selector	Static	Both	Not applicable	N/A
Screen Print Button	Static	Both	Not applicable	N/A
Screen Select Keypad	Static	Touch Screen	Not applicable	N/A
Scrolling List object: Cursor List Multistate Indicator Object List Local Message Object List Numeric Data Display Object List	Dynamic	Both	 Both Read Read Read	Retentive (if so configured)
Set Bit Cursor Point	Dynamic	Keypad	Write	Yes
Symbol	Dynamic	Both	Read	Non-retentive
Text	Static	Both	Not applicable	N/A
Time Display	Dynamic	Both	Not applicable	N/A
Trend	Dynamic	Both	Read	Non-retentive
Wedge	Static	Both	Not applicable	N/A

Objects Available in PanelBuilder

The following sections provide an overview of the different objects, and lists all the objects that are available in PanelBuilder.

ASCII Display

ASCII Display object displays a character string sent from the programmable controller. The display is updated whenever the string changes.

ASCII Input

ASCII Input object is an on-screen keyboard that allows the operator to create a text string. This text string is sent to the programmable controller.

Bar Graphs

Bar graphs display changing conditions in the process, such as temperature or fluid levels. Both horizontal and vertical bar graphs are supported.

Control Selectors

Control List Selector with Enter Key contains a vertical list of choices. An operator can use the object's **Up Cursor** and **Down Cursor** buttons to move an arrow through the available selections. The selection is sent to the programmable controller when the **Enter** button is pressed.

Control List Selector without Enter Key contains a vertical list of choices. The operator uses the object's **Up Cursor** and **Down Cursor** buttons to move through the available selections. The current selection's state value is automatically sent to the programmable controller via the object's Selector Control Tag.

Set Bit Cursor Point consists of a tag, a cursor character, and text. This object is used to "point" to screen text. Several Set Bit Cursor Points can be in the same screen. Each one can have a different (user defined) cursor character. At runtime, only the current Set Bit Cursor Point cursor character is visible and blinking. Only the current Set Bit Cursor Point's Control Tag is on (that is, 1); all other Set Bit Cursor Points' Control Tags are off (that is, 0). The programmable controller thus always "knows" the current selection. This object is available only on keypad applications.

Note: By default, when the Cancel button is pressed at the terminal, the Set Bit Cursor Point Character becomes invisible. To keep the character visible, even after the Cancel key is pressed, enable the Retain Cursor On Cancel checkbox in the Object Setup tab of the Terminal Setup dialog box. For more information about terminal setup options, see Chapter 12, *Configuring Terminal Setup Options.*

Note: If you enable the Retain Cursor On Cancel option, the application will be compatible with PanelView Version 3 or later. It will not be compatible with earlier versions.

Goto Configure Mode

The **Goto Configure Mode** object is a special object that allows operators to switch the PanelView terminal from Run Mode to Configure Mode, without using the terminal's hardware keyswitch.

The Goto Configure Mode object works with the firmware object, the Run Mode button, which is enabled separately in the firmware.

For information about the Run Mode button, and how to enable it, see the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual.*

Graphic Images

Graphic Images are imported bitmaps (in monochrome or 16-color .bmp format) that can appear as a static part of the screen or as object labels. PanelBuilder can be installed with a library of graphic images, including machine parts, symbols and arrows. You can place any of these in your application. If you wish, you can change the foreground and background colors of monochrome images, and configure them as hollow or solid.

You can also import .bmp files from another application, or AutoCAD graphics saved in the .dxf format, and place them on the screen.

Indicators

List Indicator displays a list of programmable controller states, with the current state highlighted. The value of the Indicator State control tag or expression determines which item is highlighted. State values are user-defined.

Multistate Indicator can display between 2 and 255 states, one state at a time. Each state can have a different combination of text, colors, graphic images, and attributes. The value of the Indicator State control tag or expression determines which state is displayed. State values are user-defined.

Local Message Display

Local Message Display object displays messages from the list of local messages to provide status information or instructions to the operator.

Numerics

Decrement Value Button decreases the value stored at the Button Control Tag by a whole or a fractional value each time the button is pushed. If the button is held down, the programmable controller value continues to decrease until it reaches a pre-assigned limit. The object can be configured to display the numeric value contained in the tag.

Increment Value Button increases the value stored at the Button Control Tag by a whole or a fractional value each time the button is pushed. If the button is held down, the programmable controller input value continues to increase until it reaches a pre-assigned limit. The object can be configured to display the numeric value contained in the tag.

Numeric Data Display shows the current value of the Displayed Value control tag or expression. The display can show polarity and decimal points.

Numeric Entry Keypad allows the operator to enter a value, which is then sent to the programmable controller at the address specified by the Numeric Keypad Input control. The keypad is available in large and small sizes, for touch screen applications only.

Numeric Input Cursor Point consists of a numeric display and a cursor character. The operator can enter numbers into an array of numeric fields similar to an array of thumb-wheel switches on a control panel. This object sends its control value to the programmable controller. It is available only for keypad applications.

Note: By default, when the Cancel button is pressed at the terminal, the Numeric Input Cursor Point Character becomes invisible. To keep the character visible, even after the Cancel key is pressed, enable the Retain Cursor On Cancel checkbox in the Object Setup tab of the Terminal Setup dialog box. For more information about terminal setup options, see Chapter 12, *Configuring Terminal Setup Options.*

Note: If you enable the Retain Cursor On Cancel option, the application will be compatible with PanelView Version 3 or later. It will not be compatible with earlier versions.

Numeric Keypad Enable Button pops up the Numeric Entry Scratchpad in which the operator can enter a number. The number is then sent to the Numeric Keypad Input control. This object is available only for keypad applications.

Push Buttons

Interlocked Push Buttons function as a group. When one of the buttons in the group is pressed, it cancels the other buttons and makes the selection. All the buttons in the group are added individually. All buttons use the same Button Control Tag, but each button has its own control value. The value at this tag identifies which button in the group is the currently selected option. For example, the operator can press an Interlocked Push Button to choose one motor speed out of several possibilities.

You can also use a single Interlocked Push Button to set a value in the programmable controller.

Latched Push Button changes state when pressed and remains in the changed state until unlatched by the programmable controller.

Maintained Push Button changes state when pressed and remains in the changed state when released. Pressing the push button a second time changes the button back to its original state.

Momentary Push Button (Normally Open or Closed) changes state when pressed and changes back to its Initial State when released.

Multistate Push Button sends one of a series of user-configured state values to the Button Control Tag each time the operator presses the button. You configure how many states this button has (to a maximum of 255), and assign a value to each state. Each time the operator presses the button, the value assigned to the next state is sent to the programmable controller. When the last value is sent, the procedure begins over with the first state's value being sent.

Scales

Scales consist of lines with tick-marks, used to provide an illustrative measurement scale beside Bar Graphs and Trend objects. You can create vertical and horizontal scales.

Screen Print Button

Screen Print Button allows the operator to print the application screen and pop-up windows currently displayed on the terminal.

Screen Selectors

Goto Screen Button is configured to change the screen selection to another application screen on the terminal. When the operator presses this button, the assigned screen appears.

Return to Previous Screen Button switches back to the previous screen.

Screen Keypad Enable Button pops up the Numeric Entry Scratchpad. When the operator enters a screen number, the screen with that number appears. This object is available only for keypad applications.

Screen List Selector allows an operator to change to another screen by choosing a screen from a list.

Screen Select Keypad (Large and Small) enables the operator to change to another screen by entering the screen's number. This object is available only for touch screen applications.

Scrolling List

Scrolling List objects are like scrolling versions of Control List Selectors. The number of states is not limited by the number of lines on the screen. A Scrolling List consists of a Cursor List and any number or combination of Local Message, Multistate Indicator, and Numeric Data Display Object Lists. The Cursor List can have up to 999 items. The operator or programmable controller can use the Scrolling List to control and monitor operations, or to make selections.

Shapes (Drawing Objects)

Arcs can appear in any size or shape. You can configure the arc to appear hollow or solid, and change the line style or width. You can also connect them with line objects to form rounded corners.

Ellipses are closed circular shapes. You can configure the ellipse as hollow or solid, and change the line style or width.

Lines are used to illustrate, and to separate sections of screens. Lines can be drawn in any direction, and are available in various widths and formats.

Panels are similar to rectangles, except that you can assign a border. You can also assign a hollow or solid fill.

Rectangles can surround other objects or simply illustrate. You can configure the rectangle as hollow or solid, and change its line style or width.

Wedges or pie-shapes are partial ellipses. They are similar to arcs, except they are closed. You can configure them to appear hollow or solid, and assign any line style or width.

Symbols

Symbols are like Multistate Indicator objects, except that the same monochrome bitmap appears for each state. Symbols can show up to 255 states, with user-configurable values, and different colors for each state. Any monochrome bitmap or monochrome .dxf image can be used for a symbol. You can increase or decrease the size of a monochrome bitmap to achieve the symbol size you want. PanelBuilder is installed with 32 bitmaps of standard ISA symbols, in two sizes. If the default installation was followed, these are contained in the C:\AB\PB1400E\IMAGES directory. A set of DIN symbol bitmaps is also installed there.

Text

Text is used for screen titles, to provide instructions, or to add text labels to objects. Text can be configured as transparent or solid, with background and foreground colors.

Time and Date

Date Display shows the current date. It can be displayed anywhere on the screen. Depending on the PanelView terminal settings, the object can appear in a variety of formats.

Time Display shows the current time. It can be displayed anywhere on the screen. Depending on the settings configured in the PanelView terminal, a variety of formats can be displayed.

Trends

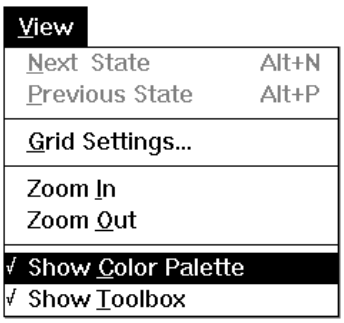
Trend allows the operator to track process variables over a period of time. It is a dynamic, graphical display object with time on the horizontal axis and the value range of the process variable on the vertical axis. Up to four process variables can be plotted using the same Trend object.

Working with Objects

Knowing how to perform several basic actions and techniques will save you time as you create objects and screens.

Using the Toolbox and Color Palette

PanelBuilder 1400e provides a toolbox for creating objects and a color palette for coloring them. The toolbox differs slightly between touch screen and keypad applications.

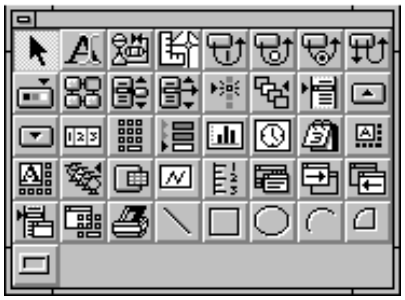


The toolbox and color palette are optional and flexible. You can:

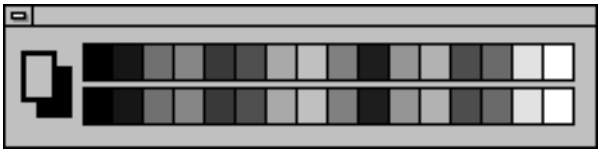
- move them anywhere on the screen, by dragging and placing them with the pointer or by using the Move command in the Control menu
- hide or show them using the commands on the View menu or on their Control menus
- resize the toolbox by dragging a side or corner, or by using the Size command in the Control menu

Touch Screen Toolbox

Keypad Toolbox



Color Palette



Selecting Objects

Before you can work with an object, you must select it.

When the Select tool is active, the pointer appears as an arrow. When another tool is active, a symbol appears below the arrow to indicate which tool is active. If the text tool is active, the pointer appears as an I-beam.

Active Select tool

Active graphic tool

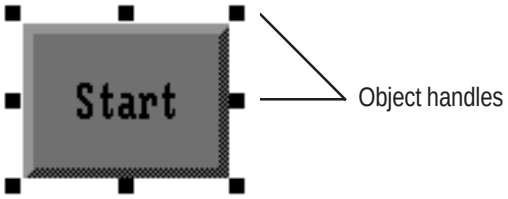
Text I-beam cursor

To choose the Select tool, do one of the following:

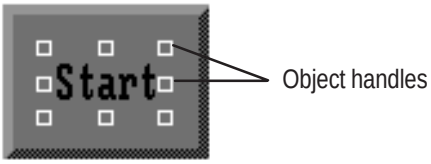
- choose Select from the Objects menu
- choose from the toolbox
- press ESC
- click the right mouse button

Selection Handles

When you select an object, handles appear around the edge of the object to show it is selected.



When you select an object that is inside of or grouped with another object, such as a grouped object or multi-component object, hollow handles appear around the edge of the object to show it is selected.



Selection Methods

To select this:	Do this:
An object	Click the object. **or** Use the arrow keys to move the selection arrow to the object and press SPACEBAR.
Several objects	Click the first object. SHIFT+click additional objects. **or** Use the arrow keys to move to the object and press SHIFT+SPACEBAR.
All objects in an area	Drag diagonally to draw a selection box around the objects. When the objects are surrounded by the box, release the mouse button. **or** Use the arrow keys to move to an object; then press CTRL+arrow keys to draw a selection box.
All objects on the screen	Choose Select All from the Edit menu. **or** Press CTRL+A.

To select this:	Do this:
An object within an object (for example, an object component or a grouped object)	Click the whole object, and then click the component or object. Hollow handles appear on the second object to indicate that it is selected. **or** Drag diagonally to draw a selection box around the components. Click the object within the first object. Hollow handles appear on the second object to indicate that it is selected. **or** Use the arrow keys to move to an object and press SPACEBAR to select the whole object; press SPACEBAR again to select the component or object. Hollow handles appear on the second object to indicate that it is selected. **Tip:** When using a selection border to select objects within another object, press CTRL+SHIFT to select only the inner objects. See the example that follows.
Overlapping objects	Click the overlapping objects, and continue clicking until you select the desired object. Each time you click, a different overlapping object is selected. **or** Use the arrow keys to move to an object and press SPACEBAR to select the object. Continue pressing SPACEBAR until you select the desired object. Each time you press SPACEBAR, a different overlapping object is selected.

Example 1: Multiple Levels of Selection

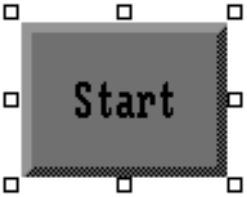
When you select an object within an object, such as one component of a multi-component object, hollow handles appear around the inner object. The status bar reports which part of the object is selected.

The following illustration shows three selection levels: the whole button, the panel, and the label. The status line is also shown.

1. Click once to select the button.

X= 15 Y= 22, Width 80 Height 60 State: 0 Normally Open Momentary Push Button

2. Click a second time to select the button's panel.



X= 15 Y= 22, Width 80 Height 60 Panel

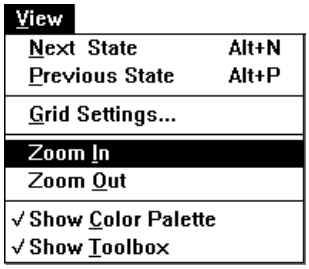
3. Click a third time to select the button's label.

X= 39 Y= 43, Width 32 Height 20 Text

Deselecting Objects

To deselect this:	Do this:
A selected object	SHIFT+click the object. **or** Press SHIFT+SPACEBAR.
Several selected objects	Hold down SHIFT and drag a selection box around the objects. **or** SHIFT+click each object.
All selected objects	Click in the drawing area, away from any objects. **or** Press ESC.

When you deselect an object, the handles around the object disappear.



Zooming In and Out

Use the Zoom In and Zoom Out commands to draw, edit, or position objects more easily.

When you first open a screen, it is in normal view. Use Zoom In to see objects in more detail. Use Zoom Out to see an overview of the entire screen, including the function keys on a keypad application.

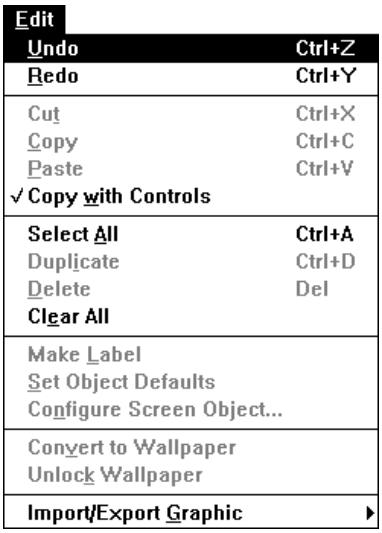
To zoom in, do one of the following:

- choose Zoom In from the View menu
- choose from the toolbar

To zoom out, do one of the following:

- choose Zoom Out from the View menu
- choose from the toolbar

If you’re at full magnification, you can reduce the view twice.



Using the Undo and Redo Commands

Use the Undo and Redo commands any time you draw or edit objects, and decide you want to reverse your actions.

When you move the pointer over the Undo or Redo icons, the status bar displays the undo or redo action that will be performed.

Important: You can undo all the operations performed in a screen since you opened the screen, or since you last saved the application. This includes configuring objects and assigning tags and expressions. One Undo resets all work done in a single session of configuring an object in the Configure dialog box (but does not undo a tag creation or edit).

To undo an operation, do one of the following:

- choose Undo from the Edit menu
- choose from the toolbar
- press CTRL+Z

When you undo an operation and then change your mind, you can reverse your action using the Redo command. You can use the Redo command as many times as you used the Undo command.

To redo an operation, do one of the following:

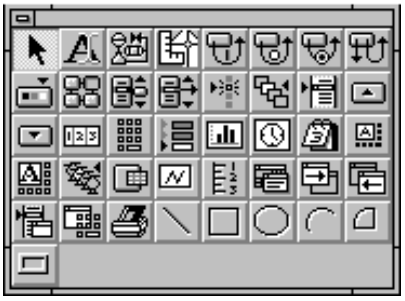
- choose Redo from the Edit menu
- choose from the toolbar
- press CTRL+Y

Object Tools and Commands

You can create dynamic and static objects by selecting tools from the toolbox or by selecting commands from the Objects menu.

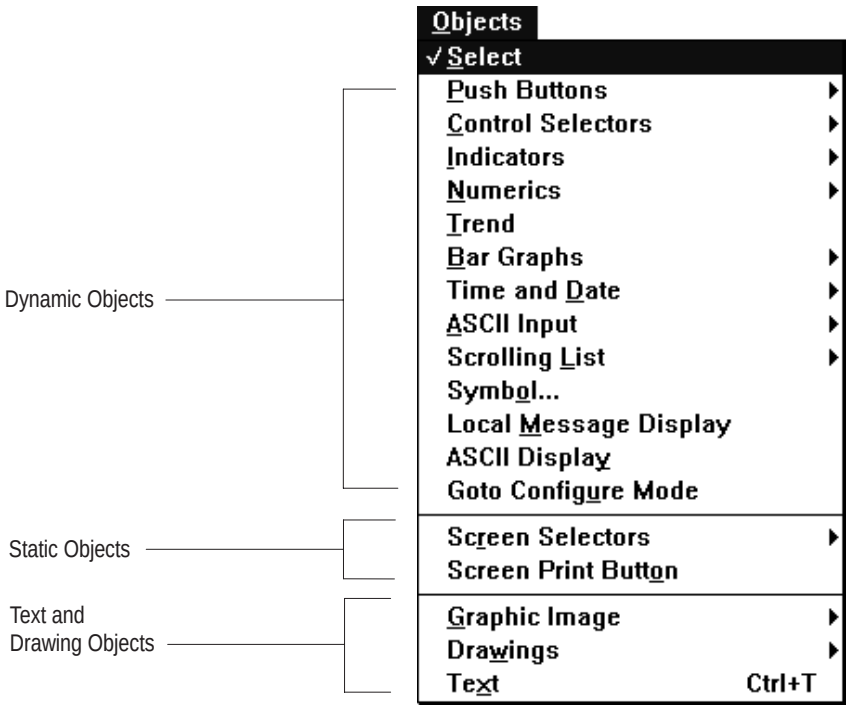
Touch Screen Toolbox

Keypad Toolbox



Note: The toolbox includes icons for the objects which are more frequently used. It does not include icons for all objects found on the Objects menu.

Objects Menu



The following table shows the object tools and their menu commands:

This tool:	Corresponds to this command:	This tool:	Corresponds to this command:
	Select		Vertical Bar Graph You can change it to Horizontal in the Configure dialog box.
	Text		Time Display
	Bitmap Image		Date Display
	DXF Image		Small ASCII Input
	Momentary N/O Push Button		Large ASCII Input
	Momentary N/C Push Button		Symbol
	Latched Push Button		Local Message Display
	Maintained Push Button		Scale
	Trend		ASCII Display
	Multistate Push Button		‘Goto Screen’ Button
	Interlocked Push Button		‘Return to Previous Screen’ Button
	Control List Selector with Enter Key		Screen List Selector
	Control List Selector without Enter Key		Screen Keypad Enable Button (Keypad applications only) Small Screen Select Keypad (Touch applications only)
	Set Bit Cursor Point (Keypad applications only)		Screen Print Button
	Multistate Indicator		Line
	List Indicator		Rectangle
	Increment Value Button You can change it to “with Display” in the Configure dialog box.		Ellipse
	Decrement Value Button You can change it to “with Display” in the Configure dialog box.		Arc
	Numeric Data Display		Wedge

This tool:	Corresponds to this command:	This tool:	Corresponds to this command:
	Numeric Keypad Enable Button (Keypad applications only) Numeric Entry Keypad (Touch applications only)		Panel
	Numeric Input Cursor Point (Keypad applications only)		

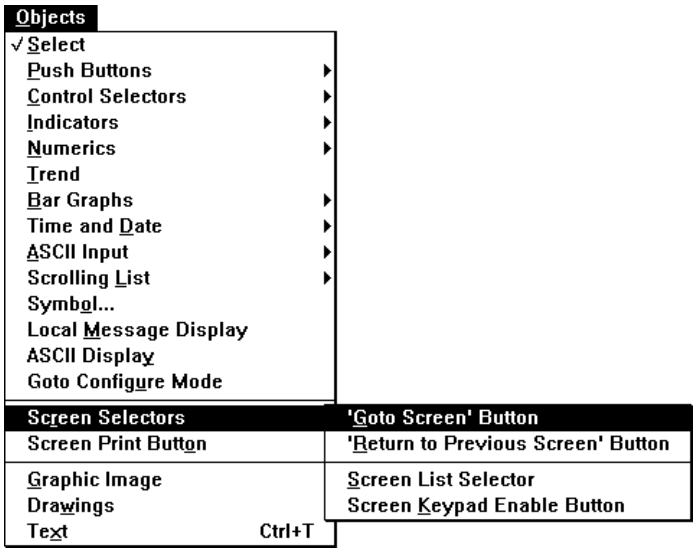
Creating Dynamic and Static Objects

The method for creating all dynamic and static objects is basically the same. The illustrations for the following procedure show how to create a ‘Return to Previous Screen’ Button.

To create an object:

1. Choose the object tool by doing one of the following:

   - choose from the toolbox
   - open the Objects menu and select the type of object to create



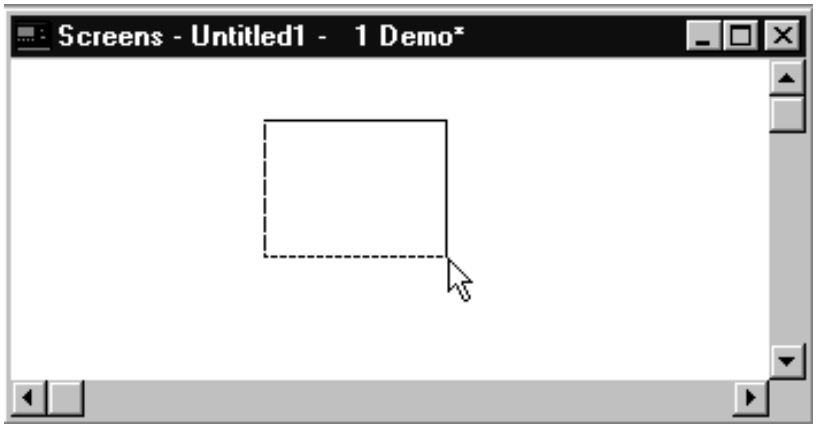
Note: Most object types have a submenu, indicated by an arrow at the right side of the Object menu. If there is a submenu, when you select the type of object the submenu appears. Then you choose the particular object.

   - For example, choose ‘Return to Previous Screen’ Button from the Screen Selectors submenu

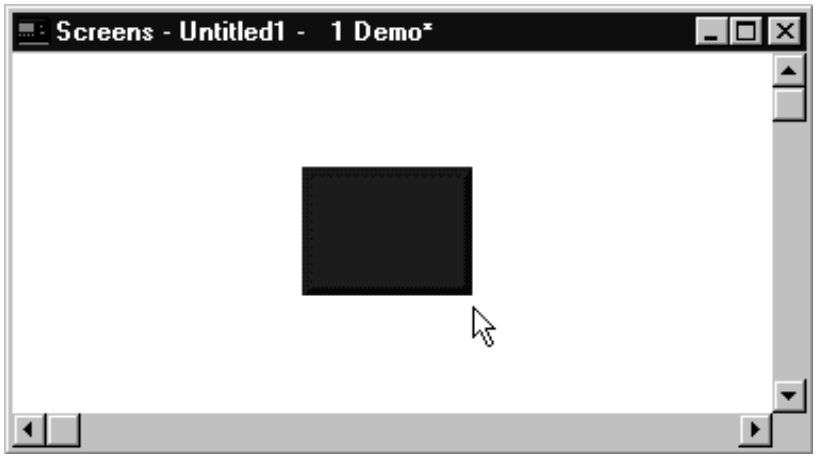
   The pointer changes to the active pointer:

2. Position the pointer approximately where you want to place the object.

3. If you're using the mouse, press the mouse button and drag the pointer until you see the object outline. If you're using the keyboard, press CTRL+arrow until you see the button's outline.



Position the outline where you would like the object and release the button.



The small square remains at the base of the pointer because the object tool is still selected. You can continue to place additional copies of the same object. If you are finished creating this type of object, deselect the tool.

To deselect a tool, do one of the following:

- click in an empty area of the screen
- press ESC
- choose Select from the Objects menu to return to select mode
- choose from the toolbox to return to select mode
- click the right mouse button
- click another object

To configure an object:

- Double-click or right-click on the object, or choose from the toolbar. A configuration dialog box opens. If you are creating an application for a keypad terminal, you assign the function keys for the object here. See the *PanelBuilder 1400e Configuration Software for Windows Reference Manual* for detailed information about configuring objects.

Creating Text

You can create background text for your screen. You can also add text to objects. You can choose the text's attributes (font, color, and so on) before you create it, or you can create the text first and assign attributes later. You can also center, left-justify, or right-justify multiple lines of text.

Note: Text lines don't wrap around. To create a second text line, press ENTER, press ↓, or click the I-beam in a new position.

When you create text inside an object as its label, it will not go beyond the boundaries of the object. If the text doesn't fit, resize the object or use a smaller font.

To create text:

1. Choose the Text tool by doing one of the following:
   - choose Text from the Objects menu
   - choose from the toolbox
   - press CTRL+T

   The pointer changes to an I-beam.
2. Position the I-beam on an empty part of the screen or inside an object, and click. To move the I-beam using the keyboard, use the arrow keys. When the I-beam is in the desired position, press any key.

   Note: If you start typing inside an object, the text automatically becomes a label. See "Adding Labels," later in this chapter.
3. Type the text.

   While the text tool is selected, you can continue creating text. Move the I-beam to another spot in the drawing area, click and type. If you're using the keyboard, press ESC and use the arrow keys to move the I-beam. Press any key and begin typing.
4. When you're finished creating text, deselect the text tool.

You can assign different foreground and background colors, font sizes, and fill styles. See “Changing Object Appearance,” later in this chapter.

Editing Text

1. Choose the Text tool as described above, or select a text string and double-click or right-click the mouse button.

   The pointer changes to an I-beam.

2. Position the I-beam on the text string you want to edit and click the mouse, or press any key and begin typing.

3. Use BACKSPACE or DELETE to delete characters.

Creating Extended Characters

In PanelBuilder you can use two extended character sets:

- PanelView extended character set—to create characters in screens and messages
- MS Word Sans Serif extended character set—to create characters in all other areas of PanelBuilder where you type text, for example, in naming screens, objects, or tags

Note: To view the extended character sets, use the Windows Character Map. The Character Map is part of the standard install set for Microsoft Windows prior to Windows 95. For Windows 95 and Windows NT, ensure the Character Map is installed.

Note: You can edit extended characters as you would edit any text string.

To use the PanelView extended character set:

1. With PanelBuilder running, open the Character Map from the Windows Accessories group.

2. In the Font field, select any PanelView font. The PanelView 8 by 20 font is preferable because it is closest to the actual pixel size of text on the screen.

   When you select a font, the characters available in that font are displayed.

   Note: The Character Map does not show all the extended characters. For a list of characters, see Appendix B, *The Extended Character Set*.

3. Select the character you want to use, and note the ALT+number key combination in the Keystroke field.

4. In PanelBuilder, open the screen where you want to create the text.
5. Select the text tool. Position the I-beam on the screen, and click the mouse button or press any key.
6. Hold down ALT and type the key combination from the Character Map on the numeric keypad.

To use the MS Word extended character set:

1. Open the Character Map from the Windows Accessories group.
2. In the Font field, select MS Sans Serif.

   The characters available in this font are displayed.
3. Select the character you want to use, and choose Select. If you wish, select additional characters.
4. When you've selected all the characters you want, choose Copy.
5. In PanelBuilder, click in the dialog box field where you want to use the character, and press CTRL+V.

Note: Instead of copying and pasting a character, you can type the character using its ALT+number key combination, as described for using the PanelView extended character set.

Creating Symbols

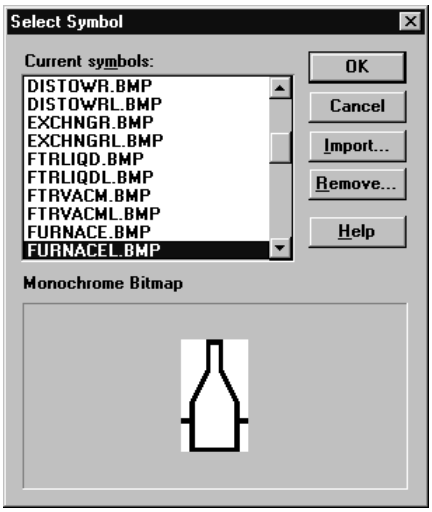
Symbols are identical in function to Multistate Indicators, except that the same graphic image label appears for every state. Symbols use less memory than Multistate Indicators, and the PanelView terminal processes information more quickly for symbols. You can include any number of symbols in your screen.

A symbol can be any monochrome bitmap drawings that you import into a screen.

Note: You can resize symbols. See "Sizing Objects," later in this chapter.

To create a symbol:

1. Open the Select Symbol dialog box by doing one of the following:
   - choose Symbol from the Objects menu
   - choose



Only imported images are available for selection. To use an image that is not in the Current symbols list, import it as described below.

2. Select an image, and choose OK. The pointer changes to the active graphic tool.
3. Position the pointer where you want to place the symbol and drag the mouse to create the object's outline.
4. When you're finished, release the mouse button.

You can assign different foreground and background colors and fill styles to monochrome graphic images. See "Changing Object Appearance," later in this chapter.

To change a symbol:

1. Double-click or right-click the symbol.

   The Configure Symbol dialog box appears.

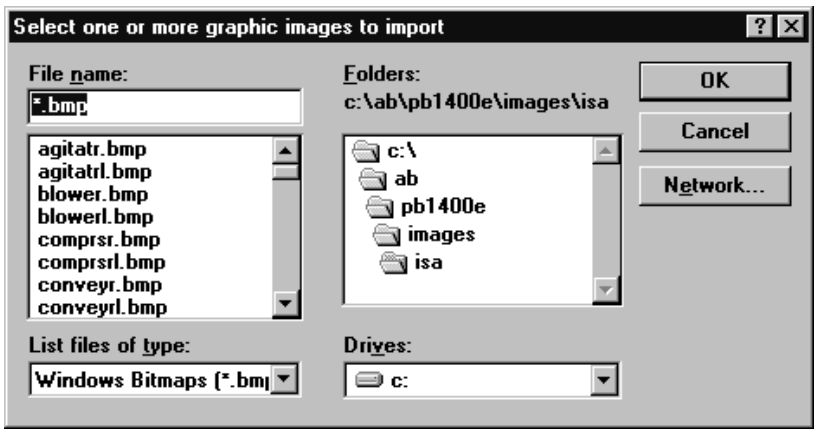
2. Click the Select Graphic Image button.

   The Select Symbol dialog box appears.

3. Select a new symbol.

To import new symbols:

1. In the Select Symbol dialog box, choose Import.



2. Select one or more files to import.

   Click a file name to select it. Use SHIFT+click to select a range of images and CTRL+click to select additional images.

3. Choose OK.

   The files are listed in the Select Image dialog box.

   Imported graphic images are stored with the PanelBuilder 1400e .pvc file. If you do not use them in any screens, remove them to keep your .pvc file to a manageable size and to conserve disk space.

To remove unused symbols:

1. In the Select Symbol dialog box, select any unused files and choose Remove.

   A warning message appears.

2. Choose Yes to remove the image.

   If you remove an image that is used in a screen, the screen will display a white box marking the spot where the image was. When you select this white box, the name of the associated file appears in the status line. You must delete the box or re-import the file. Otherwise PanelBuilder will report a validation error.

Exporting Symbols

You can export a symbol to another PanelBuilder application by simply copying it and pasting it into the new application. See “Cutting, Copying, and Pasting,” later in this chapter.

To use a PanelBuilder graphic image in a non-PanelBuilder program, import the image’s file into the program as described in that program’s documentation.

Creating Graphic Images

Graphic images are static objects, used to illustrate screens, objects, functions, and so on. You can add graphic images to the screen background, or add them to an object as object labels.

Graphic images can be monochrome or color bitmap drawings that you import into a screen. For graphic images, you can use bitmaps or AutoCAD graphics saved in the .dxf format.

Graphic images can be used as labels inside buttons. For example, you can add a graphic image for each state, so the image changes as the state changes. For details, see “Adding Labels” later in this chapter.

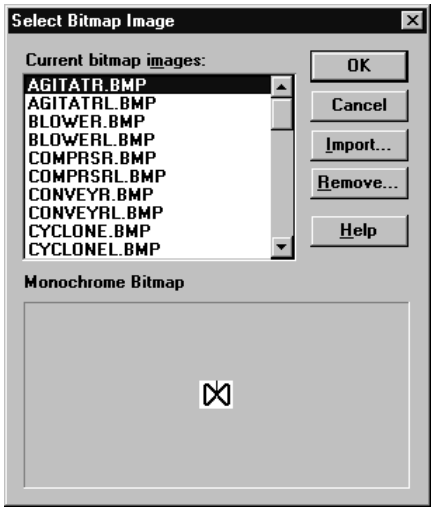
Note: You can resize graphic images. See “Sizing Objects,” later in this chapter.

Note: Not all types of .dxf images can be used in PanelBuilder. See the Readme file which accompanies your software for details about which versions of .dxf images will work with PanelBuilder.

Note: .dxf images are converted to .wmf images when they are imported into PanelBuilder. All images are converted to .bmp images when the .pvd file is written, and all images are exported as .bmp images.

To create a graphic image:

1. Open the Select Image dialog box by doing one of the following:
   - choose Bitmap Image or DXF Image from the Graphic Image submenu on the Objects menu
   - choose or from the toolbox



Only imported images are available for selection. To use an image that is not in the Current images list, import it as described on the next page.

2. Select an image, and choose OK. The pointer changes to the active graphic tool.
3. Position the pointer where you want to place the graphic image and drag the mouse to create the object's outline.
4. When you're finished, release the mouse button.

You can assign different foreground and background colors and fill styles to monochrome graphic images. See "Changing Object Appearance," later in this chapter.

To change a graphic image:

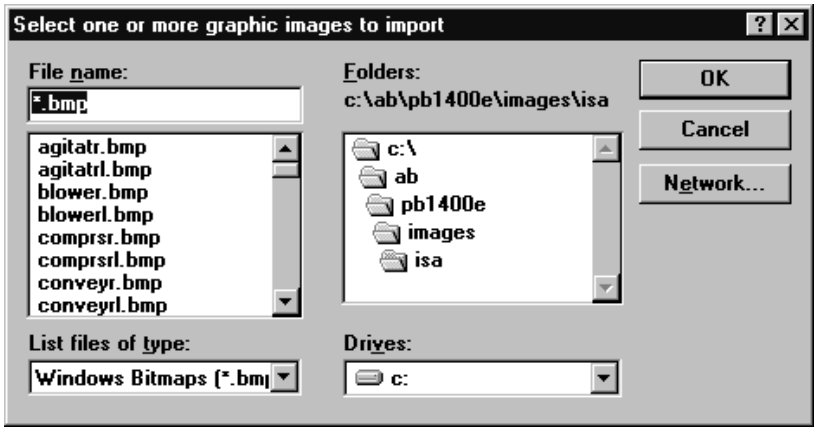
1. Double-click or right-click the graphic image.

   The Select Image dialog box appears.

2. Select a new image.

To import new graphic images:

1. In the Select Bitmap Image or Select DXF Image dialog box, choose Import.



2. Select one or more files to import.

   Click a file name to select it. Use SHIFT+click to select a range of images and CTRL+click to select additional images.

3. Choose OK.

   The files are listed in the Select Image dialog box.

   Imported graphic images are stored with the PanelBuilder 1400e .pvc file. If you do not use them in any screens, remove them to keep your .pvc file to a manageable size and to conserve disk space.

To remove unused graphic images:

1. In the Select Bitmap Image or Select DXF Image dialog box, select any unused files and choose Remove.

   A warning message appears.

2. Choose Yes to remove the image.

   If you remove an image that is used in a screen, the screen will display a white box marking the spot where the image was. When you select this white box, the name of the associated file appears in the status line. You must delete the box or re-import the file. Otherwise PanelBuilder will report a validation error.

Exporting Graphic Images

You may want to export images from your .pvd application to recreate their .bmp source files. This allows you to re-use the graphic files on other screens, in other applications, or in non-PanelBuilder programs.

You can export .bmp and .dxf images from an application (.pvc) or terminal (.pvd) file. Both .bmp and .dxf images are exported as .bmp images.

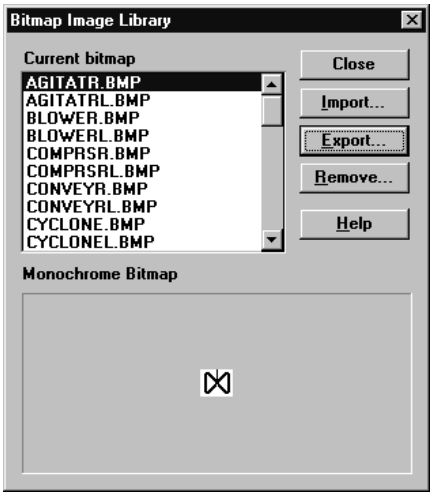
You can export a Graphic Image to another PanelBuilder application or a non-PanelBuilder program using one of the following two methods:

- copy it and paste it into the new PanelBuilder application. See “Cutting, Copying, and Pasting” later in this chapter.
- export it to a file, and import the new image file in the new application or in a non-PanelBuilder program

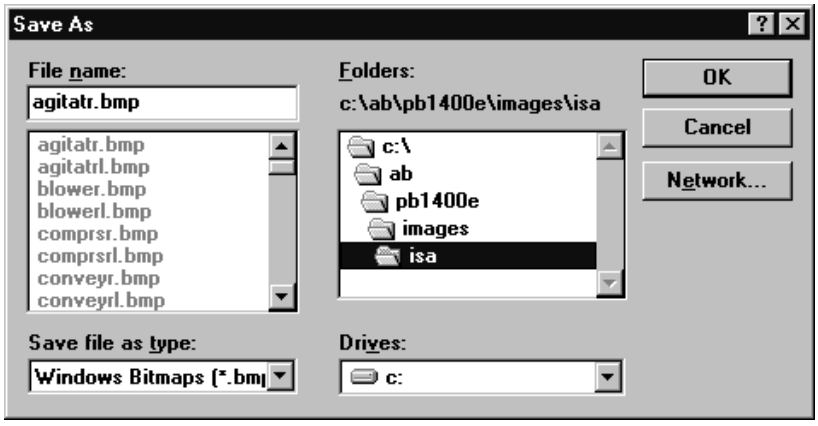
Note: To use a PanelBuilder Graphic Image in a non-PanelBuilder program, import the image’s file into the program as described in that program’s documentation.

To export new graphic images:

1. Open the Image Library dialog box by choosing Bitmap Image or DXF Image from the Import/Export Graphic submenu on the Edit menu.



2. Select an image, and choose Export.



3. In the Save As dialog box, select a path and file name for the image file and choose OK.

Creating Drawing Objects

The following table shows the tools you use to draw shapes to illustrate your screens. To create perfect circles or squares, hold down SHIFT while dragging the ellipse or rectangle tool's pointer. The table also illustrates the elements you can format for each drawing object.

Use this tool:	To draw:	Hold down SHIFT to draw:	Line	Foreground	Background	Fill	Border
Line Tool	Straight lines in any direction.	Lines that are horizontal, vertical or at a 45° angle.	✓	✓			
Rectangle Tool	A rectangle or square.	A square.	✓	✓	✓	✓	
Ellipse Tool	An ellipse or circle.	A circle.	✓	✓	✓	✓	
Arc Tool	An open segment of an ellipse or circle.	An open segment of a circle.	✓	✓	✓	✓	
Wedge Tool	A closed segment of an ellipse or circle.	A closed segment of a circle.	✓	✓	✓	✓	
Panel Tool	A rectangular object to use as a background for other objects.	A square panel.			✓	✓	✓

To draw a line, rectangle, ellipse, or panel:

1. Choose the object tool from the Objects menu or toolbox.
2. Position the pointer where you want to begin drawing.
3. If you're using the mouse, hold down the left mouse button and drag the pointer to draw the object. If you're using the keyboard, press CTRL+arrow to draw the object.
4. When the object is as you want it, release the mouse button or the keys.

See "Changing Object Appearance," later in this chapter for information about changing the object's formatting elements.

To draw an arc or wedge:

1. Choose the Arc or Wedge tool from the Objects menu or toolbox.
2. Position the pointer where you want to begin drawing.
3. If you're using the mouse, hold down the left mouse button and drag the pointer to draw the object. If you're using the keyboard, press CTRL+arrow to draw the object.

4. When the object is the right size, release the mouse button or the keys.

   The object will have four handles. When you point to one of the four handles the pointer changes to a cross.

5. Select one of the handles and drag the pointer around the circumference of the object to create the arc or wedge. If you're using the keyboard, press CTRL+arrow.

6. When the object is as you want it, release the mouse button or the keys.

You can assign different line styles, foreground and background colors, and fill styles. See "Changing Object Appearance," later in this chapter.

Adding Labels

You can add a graphic image or descriptive text to a button to inform operators of its function.

For example, you could add text labels to the two states of a push button to inform the operator whether the button is On or Off, or graphic images such as up and down arrows to show the button's purpose.

When you type text or place a graphic image inside a screen object that supports labels, the text automatically becomes a label. If you have an existing text object or a graphic image that you want to move into a screen object as a label, use the Make Label command.

Note: You cannot use PanelBuilder drawing objects (for example, arcs, lines, and ellipses) as labels.

To change a background text object or graphic image into an object label:

1. Select the background text object or graphic image.
2. Move the text or graphic image inside the screen object's borders.
3. Choose Make Label from the Edit menu or choose from the toolbar.

Note: Text or graphics must be placed fully inside the object to be a label. If text or graphics overlap the object's borders, they cannot be a label.

Note: If Make Label is dimmed, PanelBuilder automatically made the text or graphic image into a label when you placed it.

The text or graphic image is now part of the object. When you click on the text or graphic image, its handles are hollow if it is a label.

Note: For multistate objects, when you change text or a graphic image into an object label, the label is linked to the object state. You can add a different label for each object state. When the application is running and the state changes, the label changes, too. See "Configuring Object States," later in this chapter.

To remove an object label:

1. Select the text or graphic image.
2. Cut the text or graphic image. Cutting and pasting are described in the next section.
3. To keep the text or graphic image, paste it outside of the object's boundaries. This operation converts the text or graphic image back to a separate background screen object.

Editing Objects

Once you've drawn an object, you can select the object and edit it. The kinds of changes you can make depend on the type of object. You can:

- cut, copy, and paste all objects
- delete all objects
- duplicate all objects
- move all objects
- resize certain objects

Cutting, Copying, and Pasting

You can cut or copy and paste objects between screens and applications.

Note: You cannot copy keypad-specific objects to touch screen applications, or vice versa. Neither can you copy button objects or objects containing buttons between touch and keypad applications.

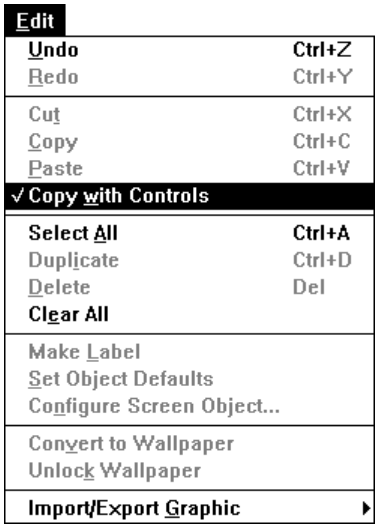
To cut or copy and paste objects:

1. In the screen, select the objects you want to cut or copy.
2. Choose Cut or Copy from the Edit menu, choose or from the toolbar, or press CTRL+X or CTRL+C.

   This places the contents in the clipboard, replacing the clipboard's previous contents.
3. Open the screen you want to copy the objects into, and click in the screen area.
4. Choose Paste from the Edit menu, choose from the toolbar, or press CTRL+V.

   The objects are pasted into the new screen.

Copying Objects with Controls



Expressions and tags control the object's inputs from and outputs to the PLC. Expression definitions and tag names assigned to an object can be copied with the object, but tag definitions cannot. If you copy controls across applications, the tags are not added to the target application. You must create the tags if they do not exist, or assign different tags.

Tip: If both applications will use the same tags, use a shared tag database.

To copy objects without controls:

- Ensure the Copy with Controls item in the Edit menu does not have a check mark (✓). If it has one, select the item so the check mark disappears.

Copying Graphic Images

If you copy graphic images across applications, the images are automatically added to the target application. If an image is already in the target application, it will be overwritten with the new one.

Deleting Objects

You can delete one or more objects on a screen.

1. Select one or more objects.
2. Choose Delete from the Edit menu, or press DELETE.

This deletes the object. If you change your mind, you can always undo your changes before closing the screen.

Duplicating Objects

The Duplicate Command is a quick way to create identical objects. Unlike the Cut and Copy commands, the Duplicate command does not use the clipboard. This command duplicates any tag names and expression definitions associated with the object.

Note: There can be only one ASCII Input object and Cursor List object per screen, so you can't duplicate either of these objects.

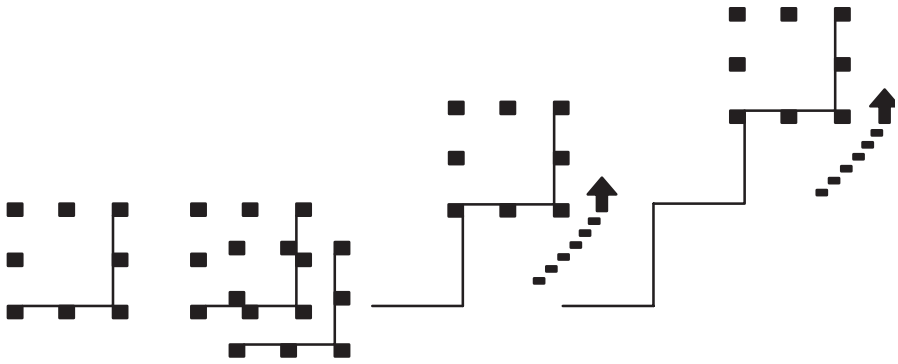
Note: There can only be four Trends per screen, so you cannot duplicate or paste the Trend more than three times.

To duplicate an object:

1. Select the object.
2. Choose Duplicate from the Edit menu or from the toolbar, or press CTRL+D.

   The duplicated object appears, slightly offset from the original object.
3. Move the new object to another position on the screen.

Tip: Duplicate also duplicates actions. For example, if you duplicate an object, move it and then duplicate it again, the Duplicate command will duplicate and move the object in a single step. This is useful if you are creating a series of objects with an equal distance between them.



Sizing Objects

You can change the size of most objects by selecting and dragging them to the size you want. The exceptions are text objects, ASCII Input object, Screen Select Keypad, and the small and large Numeric Entry Keypads.

Note: Input objects for PanelBuilder 1200e touch screen applications can be sized by touch cell increments only, not by pixels.

Note: Do not resize an object so part of it is off the screen. You cannot download an application if part of an object is off the screen.

Important: If you are sizing an object that contains a text or graphic image label, you cannot make the object smaller than the label. For example, if you size a push button that has text in state 0 and no text in state 1, you can't size the button smaller than the text size in state 0 even if you're in state 1 where you can see no text.

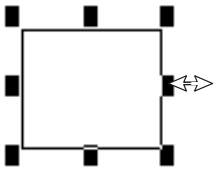
Important: If you want to resize graphic images in PanelBuilder, set your display adapter to 65,536 colors.

To change the size of an object:

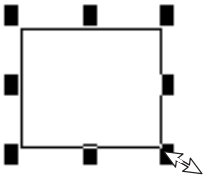
1. Select the object.
2. Position the pointer over a handle.

   The pointer changes to show the direction you can size the object.
3. Press the left mouse button. While holding it down, drag the object. If you're using only the keyboard, press CTRL+arrow to size the object. To size objects more precisely, press the + or – keys on the numeric keypad before you press the arrow key. This allows you to size the object in increments. The amount of the increment varies according to how many times you press the + or – key before you begin pressing the arrow key.

The outline of the object stretches or contracts from the anchor point to the pointer position as you move the mouse or press the keys.



Drag a side handle to change the width or height.



Drag a corner handle to change both the width and height.

4. When the object is the correct size, release the mouse button or the CTRL+arrow keys.

Moving Objects

You can move objects by using the mouse or by using the arrow keys in addition to the mouse.

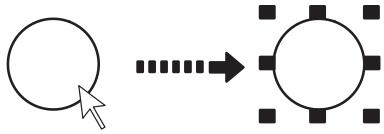
Note: Do not move objects off the screen. You cannot download an application if an object is off the screen.

Note: Input objects for PanelBuilder 1200e touch screen applications can be moved by touch cell increments only, not by pixels.

To move an object:

1. Select the object.
2. Place the pointer on the object, *not* on the handles.
3. Hold down the left mouse button and drag the object to the position you want. If you're using only the keyboard, press CTRL+arrow to move the object. To move objects more precisely, press the + or – keys on the numeric keypad before you press the arrow key. This allows you to move the object in increments. The amount of the increment varies according to how many times you press the + or – key before you begin pressing the arrow key.

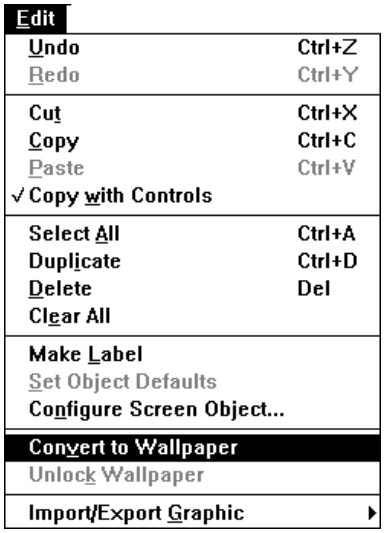
   An outline appears as you drag the object.



Select the object

Drag the object to the correct position

4. When the object is in the correct position, release the mouse button or the CTRL+arrow keys.



Converting Static Objects to Wallpaper

Wallpaper objects are part of the screen background and cannot be selected or edited. You can convert any static drawing or text object to wallpaper. Converting all static objects to wallpaper can significantly improve performance of screens at runtime.

For example, imagine you've imported a bitmap of your company logo for a screen background. The logo would work well as wallpaper. You might also select the screen's title text and other drawing objects and turn them into wallpaper.

Changing objects to wallpaper is also useful when creating screens. You cannot accidentally select objects that are wallpaper, and the drawing grid is displayed over wallpaper objects so you can easily position other objects in relation to the wallpaper objects.

To convert static objects to wallpaper:

▶ Select the objects and choose Convert to Wallpaper from the Edit menu.

To convert wallpaper back to static objects:

▶ Choose Unlock Wallpaper from the Edit menu. All objects that have been converted to wallpaper will be unlocked.

Arranging Objects

After you create objects, you can arrange them in a number of ways using tools in the toolbar or menu commands from the Arrange menu. You can:

- combine several objects into a group that behaves as a single object
- divide a grouped object into its component objects
- layer objects
- align objects with each other, horizontally or vertically
- space objects horizontally or vertically
- flip objects horizontally or vertically

Grouping and Ungrouping Objects

You can group several objects so they can be moved or edited as a single unit. You can also group an existing group with another object, or with another group. These kinds of groups are called nested groups. All groups can be ungrouped. If you ungroup nested groups, they are ungrouped in their nesting sequence.

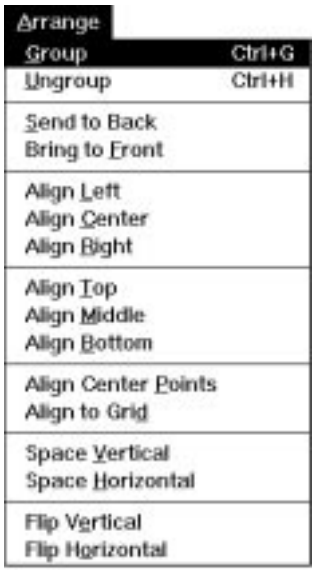
Note: Too many levels of nested groups can slow down PanelBuilder. For faster performance, keep groups to 10 or less nesting levels.

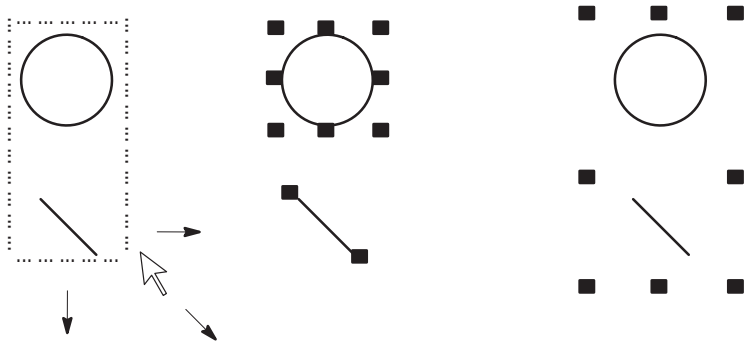
To group objects:

1. Select the objects you want to group.

   Handles appear on all objects to show they are selected. The status bar tells you how many objects are selected.

2. Choose Group from the Arrange menu or choose from the toolbar.





One set of handles appears, showing that the objects are grouped.

When you select one of these objects, you select the whole group. You can move, cut, copy, paste, or delete the group of objects in one operation, or you can change object attributes for all objects at once.

But you can still work with each object in the group individually. When you click twice on an object in a group, you select it. You can move it within the group and edit it without changing its group status. You can also cut, copy, and delete objects in the group.

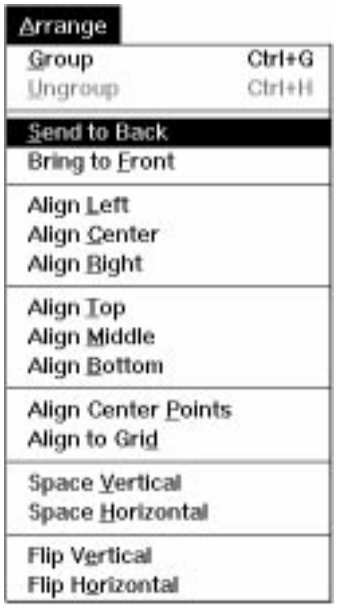
To ungroup objects:

1. Select the group.
2. Do one of the following:
   - choose Ungroup from the Arrange menu
   - choose from the toolbar

   The objects are separate again.

Editing Overlapping Objects

You can change the stacking order of overlapped objects with the Send to Back and Bring to Front commands.



Note: An input object or button can be on top of another object, but they cannot be under another object.

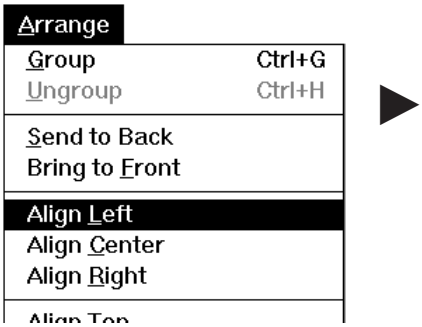
To bring an object to the front of the object stack:

1. Select the object.
2. Do one of the following:
   - choose Bring to Front from the Arrange menu
   - choose from the toolbar

Tip: To select an object that is at the back of a stack, place your pointer on the objects and click, or press SPACEBAR as many times as necessary to move through the objects in the stack. Using SPACEBAR avoids accidentally double-clicking and opening a configuration dialog box.

To send an object to the back of the object stack:

1. Select the object.
2. Do one of the following:
   - choose Send to Back from the Arrange menu
   - choose from the toolbar



Aligning Objects

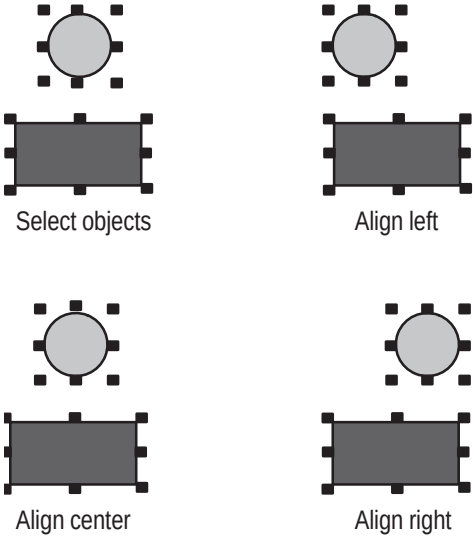
Use the commands in the Arrange menu to align the objects in several ways.

Note: Input objects for PanelBuilder 1200e touch screen applications automatically snap to the touch cell grid (64 by 40 pixel grid).

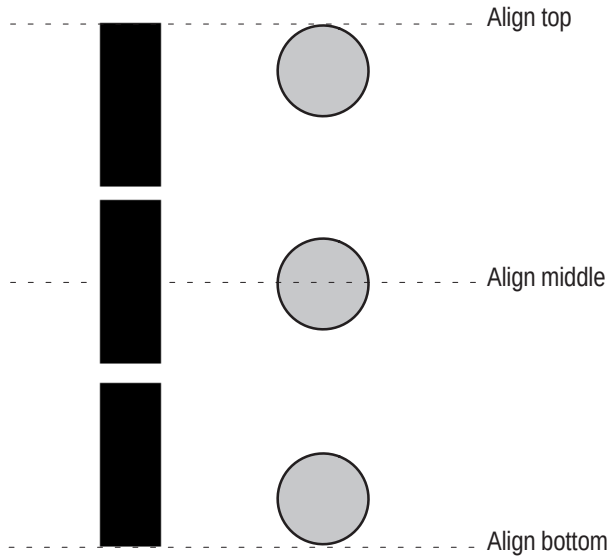
1. Select the objects you want to align.
2. Choose a command from the Arrange menu to align the objects in relation to each other.

This command:	Aligns selected objects to this object:
Align Left	Left-most selected object
Align Center	Horizontal center of all selected objects
Align Right	Right-most selected object
Align Top	Top-most selected object
Align Middle	Vertical center of all selected objects
Align Bottom	Bottom-most selected object
Align Center Points	Center of all selected objects
Align to Grid	Aligns all selected objects' upper left-hand corners to the nearest screen gridpoint

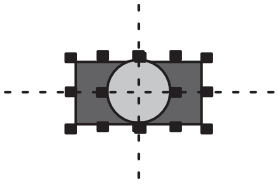
Aligning objects left, right, and center



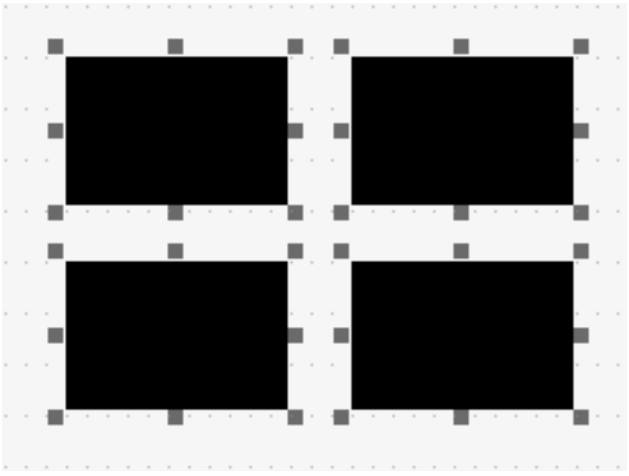
Aligning objects top, middle, and bottom



Aligning center points of objects



Aligning objects to grid

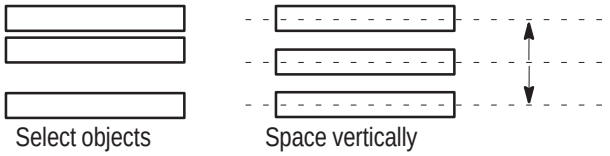


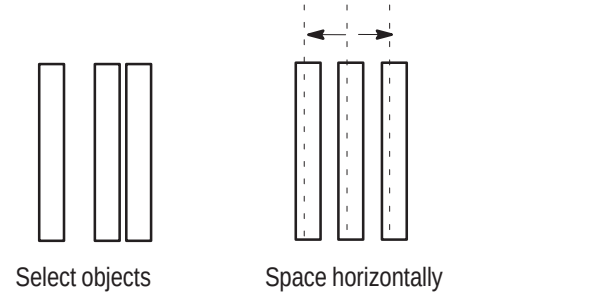
Spacing Objects

Use the Space Vertical and Space Horizontal commands to space the objects vertically or horizontally, so the center points of the objects are at equal distances from each other.

1. Select the objects you want to space.
2. Choose a space command from the Arrange menu.

This command:	Does this:
Space Vertical	Places the centers of the selected objects an equal distance apart vertically.
Space Horizontal	Places the centers of the selected objects an equal distance apart horizontally.





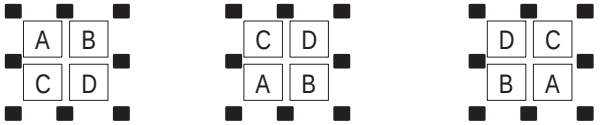
Flipping Objects

Flipping a group of selected objects creates a mirror image of those objects. You cannot flip single objects, such as graphic images, symbols, buttons, lists, and text objects.

1. Select the objects or object components.
2. Choose a flip command from the Arrange menu.

This command:	Flips selected objects this way:
Flip Vertical	Top to bottom (upside-down)
Flip Horizontal	Left to right

The following illustration shows objects flipped vertically and horizontally.



Select group Flip vertical Flip horizontal

Arranging Labels

You can arrange labels (such as multiple lines of text or graphic images) within an object. To space labels vertically or horizontally, you must first position one object at each end of the area in which you want to arrange them.

To align or space multiple lines of text within an object:

1. Position the two outermost text objects as desired.
2. To select only the text elements, press CTRL+SHIFT while you draw a selection border.
3. Choose the appropriate Align or Space command.

You can also arrange the text relative to the object. For example, you could arrange the text to line up with the sides of the object or the middle.

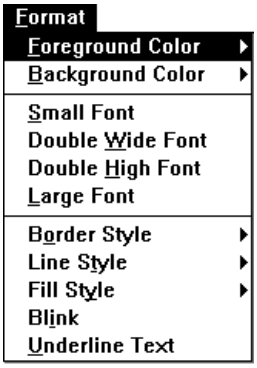
To arrange text relative to the object:

1. Select all the text objects. Then SHIFT+click the object to select it.
2. Choose the appropriate command (for example, Align Center) to arrange the text as you want.

Changing Object Appearance

Each object has certain default settings for colors, size, number of states, and so on. You can change these settings to suit your needs.

You can apply different foreground and background colors to objects, select different borders, or add emphasis to objects by making them blink.



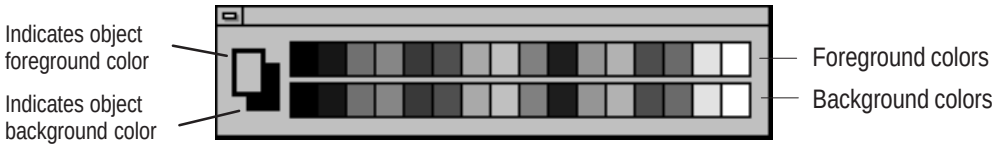
Setting the Object's Foreground and Background Colors

You can set different foreground and background colors for graphic objects and screen objects. For static objects, choose the colors before you create the object. For dynamic objects, create the object and then specify the colors.

Dynamic objects have default appearance attributes. For details about viewing the defaults, see "Configuring Object States," later in this chapter.

You can set an object's colors by using the Foreground Color and Background Color options in the Format menu, or by using the color palette.

The color palette consists of two identical color bars. The colors in the upper bar are the foreground colors. The colors in the lower bar are the background colors.



If multiple objects are selected whose foreground colors differ, the foreground color indicator in the color palette will be displayed as a dotted rectangle. Likewise, the background color indicator will appear as a dotted rectangle if the background color of the selected objects differ. The color menu will show these conditions by not displaying a check (✓) beside any color.

Foreground colors are used for text, monochrome graphic images, lines, arcs, wedges, ellipses, rectangles, bar graph fill, scale lines, and for button and list highlight colors. Background colors are used for the filled part of the rest of the object.

How Colors Appear at Runtime

For buttons and button objects, the foreground color of the button's panel determines the color of the button border highlight when the button is pressed.

For list objects, the foreground color of the list's panel is the highlight color when a state in the list is selected.

To apply colors before drawing:

1. Select a drawing tool or the text tool.
2. Select the colors from the color palette or from the Format menu.

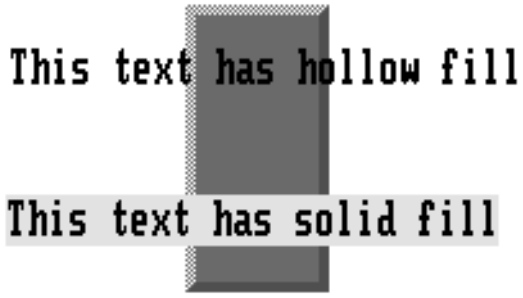
Tip: If you're drawing static objects and want to use the attributes of an existing object, select that object first. Then choose the object tool you want and begin drawing or typing. The new object will have the same attributes as the selected object.

To apply colors to an object:

1. Select one or more objects.
2. Select the colors from the color palette or from the Format menu.

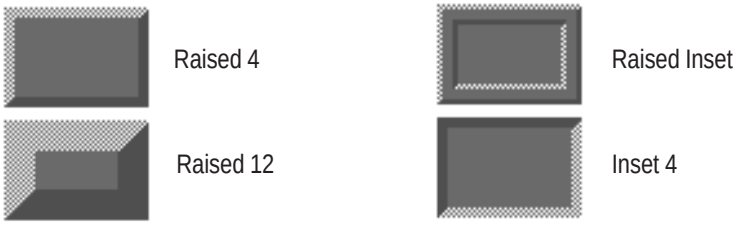
Setting an Object's Fill Type

Many objects can be hollow or solid. If you make an object hollow, any object that it covers will show through. The following illustration shows the difference between hollow and solid fill when applied to text.

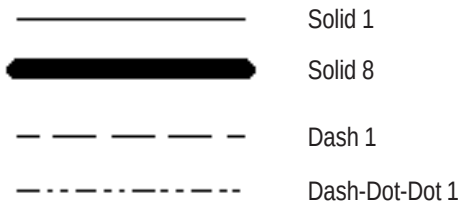


Changing the Border and Line Styles

You can change the border style of most objects. The following panel objects show four possible border styles:



For line, ellipse, arc, wedge, and rectangle objects, you can choose a variety of line styles. The following illustration shows four possible styles:



To change an object's border or a line's style:

1. Select the object or line.
2. Choose a border or line style from the Format menu.

Making Objects Blink

One way to add emphasis to objects is to make them blink. Blinking alternates between the object's foreground or background color and the background screen. For example, a panel blinks between its background colors and the screen color. A text object blinks between its foreground color and the screen color. Do not make too many objects blink because blinking objects can be distracting.

1. Select the object.
2. For a multistate object, select the desired state.
3. Choose Blink from the Format menu.

The object won't blink on your screen, but will blink when the application is running in the PanelView terminal.

Changing Text Appearance

You can change the font size to one of these options:

Small Font

Double Wide Font

Double High Font

Large Font

Extra Large Font

You can also choose Underline Text to have the text underlined.

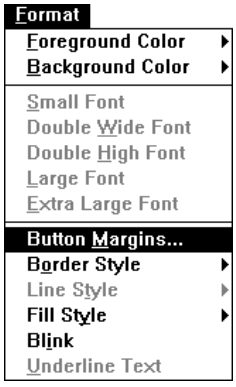
1. Select the text object.
2. Choose a text size or choose Underline from the Format menu.

   The text will change according to your selection.

Note: Text labels cannot exceed an object's boundaries. To change to a larger font that would exceed the boundaries, select the object first, then increase the text size. The object will automatically resize to accommodate the larger text.

Tip: To change the size of several text objects at once, select them all and apply the new size.

Changing Touch Margin Width or Height (Touch Screen Only)



In a touch screen application, you can change the touch margin width or height for buttons. Using a large touch margin prevents users from accidentally pressing an adjacent button.

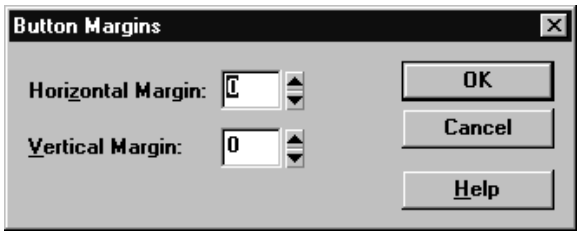
The margin width or height increases the size of a button, so the button overlaps adjacent touch cells. When the application is running, only the middle area of the button is touch-sensitive—the touch cells covered by the margin are not touch-sensitive.

Important: Do not allow the button margin to overlap other buttons or button margins, or to be positioned off the screen. An application cannot be downloaded if any buttons overlap or are off the screen.

To specify the size of the button margin:

1. Select the button.
2. Choose Button Margins from the Format menu.

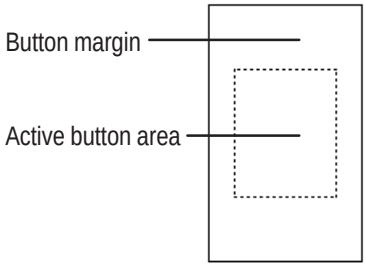
   The Button Margins dialog box opens.



3. Specify horizontal and vertical margins for the button. The numbers you specify determine the number of pixels the margins will occupy.

   The button will change to the specified size. Although the button appears larger on the screen, only the original area is touch-sensitive.

   When you select and move or size the button, a dotted rectangle appears around the active touch area. The rectangle indicates the button's touch area. The solid outer rectangle indicates the margin area, and changes size as the button is stretched or moved.



Working with Multi-Component Objects

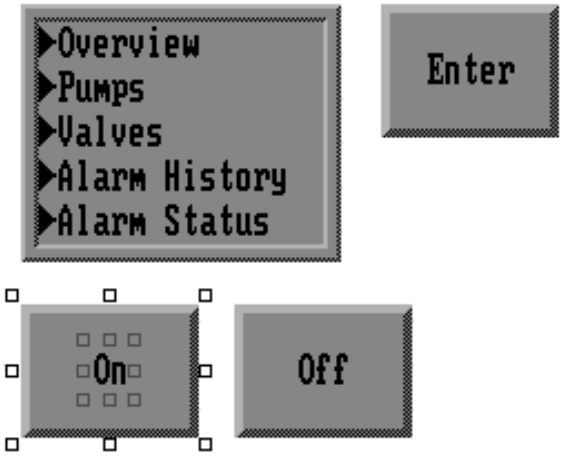
Some objects, such as Control List Selectors and ASCII Input objects, consist of multiple components. Depending on the object, you can change the format attributes of the different components. For example, you can change border styles, touch margins, fill type, and foreground and background colors. You can also resize many components, or move them.

To edit a component of a multi-component object:

1. Select the main object by clicking it.
2. Click again on the component.

For details on how to select the various components of a multi-component object, see “Selecting Objects,” earlier in this chapter.

Hollow handles appear around the component. You can edit it according to the options available from the Format menu.



The button is selected, and the text “On” is selected.

Setting Object Defaults

An easy way to create similar dynamic objects is to use the Set Object Defaults command. You create one object, edit it as you want, and then use the Set Object Defaults command to designate this object’s attributes as the default attributes for all new objects of that type.

Note: Static drawing objects, labels, function keys, and tags cannot be defaults.

1. Create the object whose defaults you want to assign.
2. Edit the object as you want. You can change the line or border, colors, and size. If the object can have several states, you can configure the different states as defaults. However, you cannot add text as a default.
3. Choose Set Object Defaults from the Edit menu.

   Every new object of this type you create in this application will appear like the one you just created.

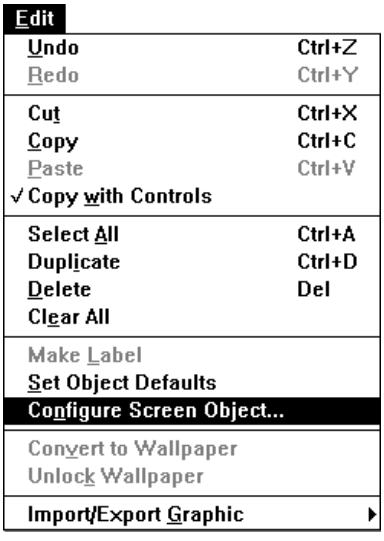
Important: The defaults you assign will apply to all applications you create and to all applications you modify. If you want to create applications that use the original defaults, before you assign new defaults you should exit PanelBuilder and rename the defltgfx file in the \PB1400e directory, or copy it to another directory. To use the PanelBuilder defaults again, rename the file to defltgfx or copy the file back to the \PB1400e directory before opening PanelBuilder 1400e.

Configuring Objects

You can configure all dynamic objects and some static objects in the object's configuration dialog box. The configuration information differs from object to object. The *PanelBuilder 1400e Configuration Software for Windows Reference Manual* explains each object in detail and describes how to configure them. This section describes only the most common types of configuration. These are:

- assigning function keys to input objects on keypad applications
- configuring object states
- configuring list object states
- assigning tags and expressions to objects

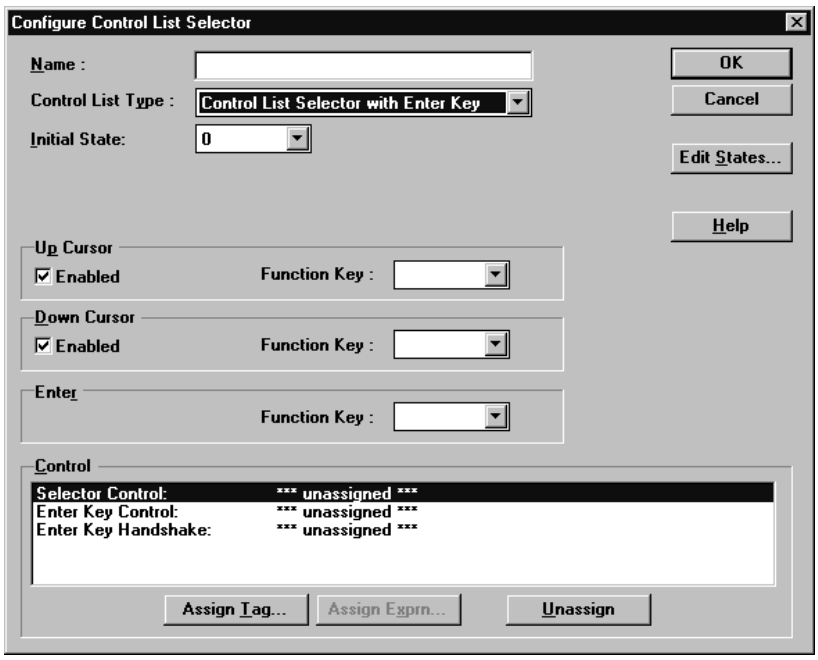
Note: To minimize the size of the application file, do not assign names to objects. Names use one byte per character.



To open an object's configuration dialog box, do one of the following:

- select the object and choose Configure Screen Object from the Edit menu
- select the object and choose from the toolbar
- select the object and right-click the mouse button
- place the pointer over the object and double-click
- place the pointer over the object and press ENTER

The options you can configure differ from object to object. The following illustration shows the Configure Control List Selector dialog box. Here you can choose between the two different types of Control List Selectors, define different state values, configure the object's buttons, assign function keys to them, and assign tags and expressions.



Assigning Function Keys (Keypad Applications Only)

Operators use the function keys on PanelView keypad terminals to control keypad applications. Instead of pushing a button on the screen to carry out a command, the operator pushes the function key that has been assigned to that button.

The best way to ensure that the operator presses the correct function key is to create a label for the button and then link the button to a function key. For example, if you want the operator to press F21 to go to the next screen, create a “Goto Screen” button, and position it on the screen next to the F21 key. Then create a label that says “F21” and assign F21 to the button.

▶ **Tip:** To see which objects have been assigned function keys, choose Reports from the application window’s File menu, and print a report on the screen using the Report Selection dialog box. Use the View menu’s Zoom Out feature to see the function key layout on the terminal.

Important: Assign the same function key only once for each screen.

To assign a function key:

1. Select a button or create a new button.
2. Open the object’s configuration dialog box using one of the methods described previously.
3. From the Function Key drop-down list, select the function key you want to assign. Repeat for each button in the object.

Function keys that are already assigned have an asterisk (*) beside them. Function keys that have more than one assignment have multiple asterisks. Do not assign the same function key more than once in a screen.

Tip: If you need to use two function keys simultaneously, use one key from the horizontal group (F1 to F16) and one from the vertical group (F17 to F21). Do not use two keys from the same group.

4. Choose OK.

Tip: Where appropriate, move the button so it is positioned next to the assigned function key. This makes the application easier to use on the PanelView terminal.

To unassign a function key:

- Select the blank item in the Function Key drop-down list, or assign another function key to the selected button.

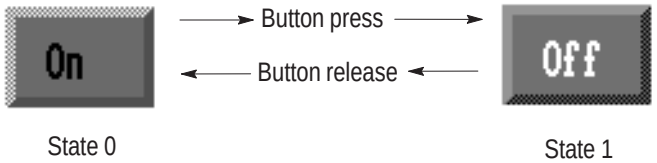
Configuring Object States

Certain objects in PanelBuilder 1400e can have different states. There are two-state objects and variable-state objects.

These are the two-state objects:

- Momentary Push Buttons
- Latched Push Buttons
- Maintained Push Buttons

Each time a two-state button is pressed, it changes state. For example, when a Normally Closed Momentary Push Button is pressed, it changes from On (State 0) to Off (State 1).



You can change the appearance of the object for each state. For example, if State 0 of a Normally Closed Momentary Push Button appears black and displays the word “On”, for state 1 you might change the color to white and add the word “Off”.

These are the variable-state objects:

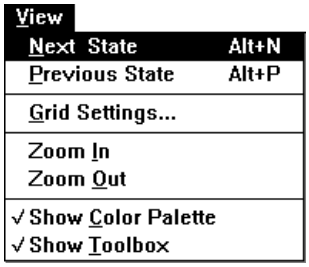
This object:	Can have this many states:	Is each state's appearance individually configurable?
Multistate Indicator	2 – 255	Yes
Multistate Push Button	2 – 255	Yes
Scrolling List's Multistate Indicator Object List	2 – 255	Yes
Symbol	2 – 255	Yes

The table shows that multistate objects can have a different appearance depending on which state is active. You can individually configure the appearance of each state. When you create a new object, each state's default appearance is already assigned.

To view the default state appearances:

1. Select the object.

   The status bar shows which state is being displayed.



2. Select the object state to view by doing one of the following:
   - choose Next State or Previous State from the View menu
   - press ALT+N for next state or ALT+P for previous state
   - select

from the toolbar (by using the mouse or by pressing CTRL+SHIFT+S) and choose a state

   The appearance of the object changes. The status bar indicates the object's new state.

3. If there are more states, move through them to view them all.

To configure state appearances:

1. Select the object.
2. Select a state as described above.
3. Make the desired changes using the options from the Format menu. If you wish, add text or graphic image labels.

Configuring Number of States

Variable-state objects are created with a default number of states. For example, a symbol has four states by default. You can change the default number of states. The table on the previous page shows the allowable number of states for each variable-state object.

To configure the number of states:

1. Open the object's configuration dialog box using one of the methods described previously.
2. For variable-state objects, except Control List Selectors, assign the number of states here:

For Control List Selectors, assign the number of states by sizing the list, as described in "Configuring List Object States," later in this chapter.

Note: When you increase the number of states, the attributes of the last state are used for all new states. You must assign state values for these new states.

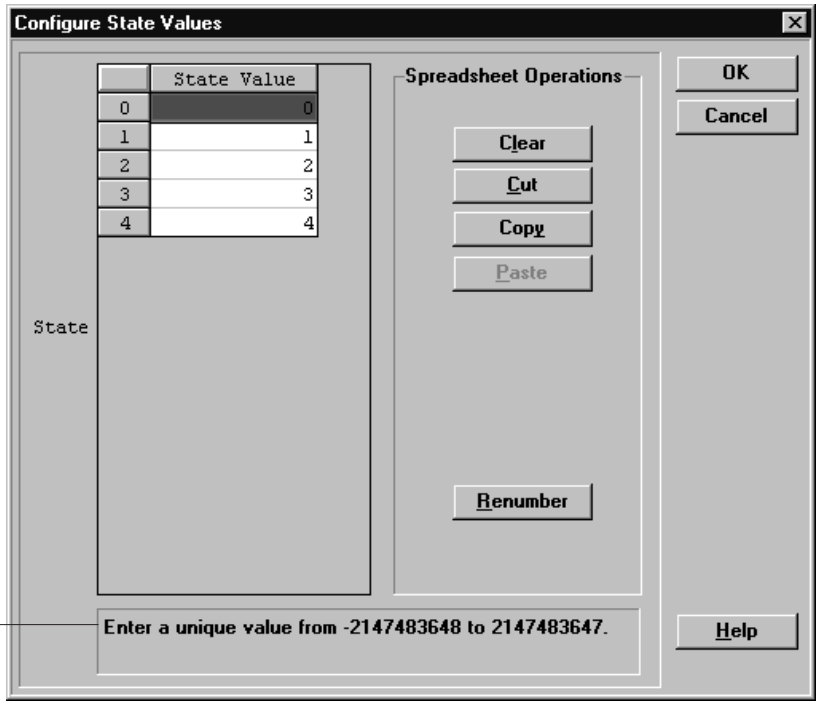
Assigning Values

You must assign a value to each of the object's states.

Note: For the Scrolling List Object's Cursor List, you can also specify whether to skip individual states. See the *PanelBuilder 1400e Configuration Software for Windows Reference Manual* for more information.

To assign state values:

1. In the object's configuration dialog box, choose Edit States. The Configure State Values dialog box opens.



The state value range is a long signed integer value.

2. Click on the state to which you want to assign a value.
3. Type the numeric value for the state.

   For input objects, this value is written to the programmable controller when the object's state changes. For example, with a Multistate button, each time the operator presses a button the state changes to the next state and the state value is written to the Button Control tag. For output objects, this value determines which state appears on the screen.

   Important: To minimize the size of the application file, use state values that increment by one for each state. Do not configure more states than you need.

   Note: For analog tags, the values are descaled before they are written to the PLC, and are scaled when they are read from the PLC. The following formulas are used:

   Descaled Value = (Value – Offset) / Scale
   Scaled Value = (Value × Scale) + Offset

If the state values are to be read from or written to an analog tag, they must be scaled values. For details see Chapter 6, *Defining Tags*, and Chapter 7, *Creating Expressions*.

To remove a state value:

- Select the state value and choose Clear.

To renumber state values:

1. Click the first state value and drag the mouse to highlight all the state values to renumber.

   Tip: To quickly select all state values, select the State Value column heading, or select the first state value and then press CTRL+SHIFT+END.

2. Choose Renumber.

   PanelBuilder sequentially assigns state values to all selected states using the first highlighted state as the starting value. State values are assigned in increasing order.

Error States

If an object has more than one state, these two conditions can cause an error state:

- The value of the object's control does not match any of the state values defined for the object.
- The control for the object has not been assigned.

For button objects, if an error state occurs, the highest state is displayed without its label (for Interlocked Push Buttons, no button is highlighted). For list objects, if an error state occurs, the highlight bar disappears.

Configuring List Object States

You can configure these types of list objects:

- List Indicators
- Control List Selectors
- Screen List Selectors
- Scrolling Lists

At the PanelView terminal, the operator moves through the items in the list and makes selections. Each item in the list is a different object state.

You can define the number of states in the list by resizing the list component.

Note: You can also define the number of states for all but the Control List Selector in the objects' configuration dialog boxes.

Note: For the Scrolling List object, the number of states in its Object Lists depends on the number of visible states defined for its Cursor List. Once you've assigned tags to visible states in the Scrolling List, you can't remove the states by resizing the list. You must first unassign the visible state tags.

Important: If a list object has state text defined, you can't resize the list so it is smaller than the number of states with text. You must delete the text before you can resize the list.

To define the number of states in the list:

1. Select the list component of the list object.
2. Resize the list component vertically, until it shows the desired number of states.

   Note: The maximum number of states an object can display depends on the font size.

If you select this font size:	The object can display this many states:
Small or Double Wide	24
Double High or Large	12
Extra Large[1]	6

[1] You cannot use the Extra Large Font for Scrolling List objects

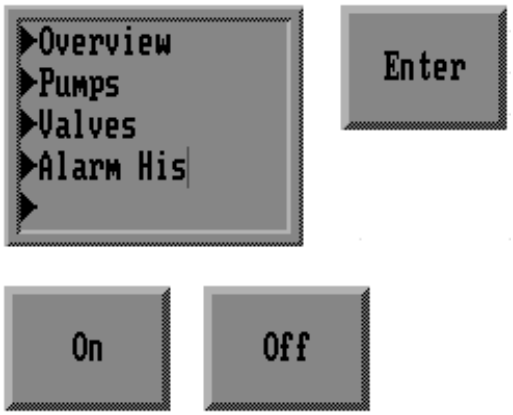
You're now ready to configure the states. For example, if the object is a Screen List Selector, you can assign a different screen to each state. See the *PanelBuilder 1400e Configuration Software for Windows Reference Manual* for more information.

Adding Text to List Object States

When you've defined all the states for your list, you can add text to the list. This allows the operator to select states from the list.

The object in this illustration is a Screen List Selector.

The Selector's list has five states, as indicated by the arrows. You can add text for each state in the list.



To add text to each state:

1. Select the Text tool.
2. Position the I-beam next to the first state in the list and type the name for that state.
3. Move to the next item in the list and type the name for the state. The quickest way to move to the next item is to press ENTER.

Note: While you edit text, the status bar indicates the current state.

Configuring Initial States

Certain objects have pre-defined initial states, while other objects allow you to configure the inital state.

You can configure the initial state for the Control List Selectors, the Maintained Push Button, and the Multistate Push Button.

When you assign a tag to a control, the tag you assign to the object can also have an initial value defined for it. If you define both an initial state (for an object) and an initial value (for a tag), the object's initial state will take precedence.

If you leave the Initial State field blank, the tag's initial value is used instead of the state value.

To specify the inital state:

1. Open the object's configuration dialog box using one of the methods described previously.
2. Enter a value in the Initial State field.

Important: A tag's address can have only one initial state value. If you assign more than one initial state value to a tag address (by assigning the same tag to two different objects with different initial states), an error will be reported during the application validation process.

Assigning Tags and Expressions to Objects

You must assign tags or expressions to all dynamic objects on your screens. For information about tags, see Chapter 6, *Defining Tags.* For information about expressions see Chapter 7, *Creating Expressions.* For details about a particular object, see the *PanelBuilder 1400e Configuration Software for Windows Reference Manual*—check the configuration section for the object you are configuring.

All dynamic objects require at least one tag or expression. Depending on the object, several tags and expressions may be required. In general, for read functions you can use either a tag or an expression. The only exception is the ASCII Display object, which uses a string tag. Expressions do not support string tags.

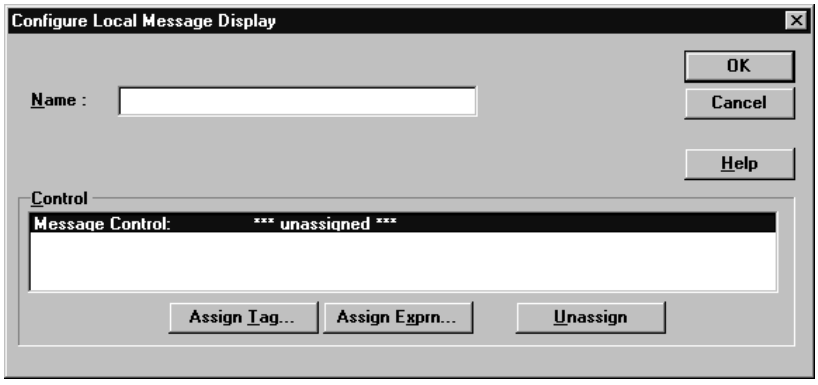
For write functions, you generally must use tags only. The exceptions are the Numeric Entry Keypad, Numeric Input Cursor Point, and Numeric Keypad Enable Button objects. These objects allow you to use an optional write expression that manipulates the operator's input before sending it to a write tag.

You can define each tag when needed using the Tag Browser dialog box, or you can define all the tags in the Tag Database editor. You define expressions individually in the configuration dialog box for each object (although you can copy expressions from one object to another).

To assign a tag or expression:

1. Open the object's configuration dialog box using one of the methods described previously.

   The following illustration shows the configuration dialog box for the Local Message Display object. This object uses only one control, named Message Control.



2. To assign a tag, double-click the control name, or click the control name and choose Assign Tag.

   The Tag Browser dialog box appears. From this dialog box you can choose an existing tag (by double-clicking it) or create a new one (by choosing New Tag).

   If you choose New Tag, the Tag editor appears. For details on using the Tag editor, see "Other Methods of Creating Tags" in Chapter 6, *Defining Tags*.

3. To assign an expression, click the control name and choose Assign Exprn (Expression). This button is dimmed if you select a control to which only tags can be assigned.

   The Expression editor appears. For details on using the Expression editor see Chapter 7, *Creating Expressions*.

To unassign a tag or expression:

1. Open the object's configuration dialog box using one of the methods described previously.
2. Click the control name and choose Unassign.

If you unassign a tag, the tag remains in the database, but is no longer assigned to the option. If you unassign an expression the expression is lost.

Viewing Assigned Tags

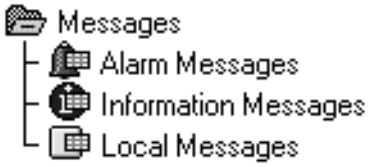
You can view the tags used in a screen by selecting Screen Tag Usage Viewer from the Screen menu, or by selecting Tag Usage Viewer from the Application Window's Tools menu. For details, see "Using the Tag Usage Viewer" in Chapter 6, *Defining Tags*.

Creating Messages

This chapter tells you about:

- the different kinds of messages
- how messages are triggered
- creating and editing messages
- using formatting codes in your messages
- adding options to alarm messages
- importing and exporting messages
- validating and updating messages

What Are Messages?



Messages keep the operator informed about the status of the operation when the application is running. Messages provide the operator with prompts, instructions, warnings, and information on current status and about what is happening in the process.

Kinds of Messages

You can create messages in each of the Messages editors. There are three kinds of messages:

- **Alarm messages** alert the operator to problems in the process. Each alarm message can be equipped with special notification options. These can sound an audio alarm, print the alarm message, display the message in the Alarm Window on the terminal's screen, send the alarm message to a PLC, and energize an alarm relay. You can create a maximum of 2000 alarm messages.

  Alarm messages appear in the single-line, multi-line, and panel Alarm Windows on the PanelView terminal's screen, and in the Alarm History and Alarm Status screens.

- **Information messages** inform the operator about the status of the operation, providing prompts, instructions, or warnings. You can create a maximum of 2000 information messages.

  Information messages appear in the Information Message Window on the PanelView terminal.

- **Local messages** inform the operator about the status of an operation or provide prompts or instructions. You can create a maximum of 2000 local messages.

  Local messages appear in the Local Message Display object in application screens. The list of local messages is used by all the Local Message Display objects in all the screens of the application. You can also configure Local Message Displays as part of the Scrolling List object.

For information about the Alarm and Information Message Windows, see Chapters 11 and 12. For detailed information about the Local Message Display object, see the *PanelBuilder 1400e Configuration Software for Windows Reference Manual.*

How Messages Are Triggered

Each message is assigned a trigger value. This value must be unique in each message editor, and can be any signed long integer except 0. At runtime, messages are displayed according to their trigger values.

For example, when the programmable controller moves a number that matches an information message trigger value to the Information Window Control, the Information Message Window appears (if it is enabled) and displays the message that has been assigned this trigger value.

The trigger values you assign can help simplify PLC programming. For example, you could assign trigger values to these messages as follows:

Message	Trigger Value
Temperature is –32°	–32
Warning—Temperature is 1009°	1009
Danger! Temperature is 1457°	1457

In this example, the trigger value of each message matches its temperature.

Creating and Editing Messages

You create and edit each message type in its own editor.

Note: The procedure for creating and editing alarm, information and local messages is identical, except that alarm messages have extra features. These features are explained separately.

Guidelines for Creating and Editing Messages

- Each message must have both a trigger value and text. Alarm messages are the only exception, in that text is optional. If the audio, print, or relay alarm option is set to Yes, you don't need to enter text. But if the Message to PLC alarm option is set to Yes, you must enter text.
- The trigger value must be a signed long integer. A value of 0 cannot be assigned.
- Each trigger value must be unique in each editor.
- You can use any ASCII character, including the characters in the extended ASCII character set. See Appendix B, *The Extended Character Set*. Also, *ASCII Displays* in the *PanelBuilder 1400e Configuration Software for Windows Reference Manual* lists the regular ASCII character set.

The following table describes the fonts and message lengths you can use:

	Local Messages	Information Messages	Alarm Messages
Font size	all	all except Extra Large	Single-line window: Small Multi-line window: Small and Double High Panel window: all except Extra Large
Where to configure font size	Format menu (with the Local Message Display object selected)	Configure Information Message Window dialog box	Message Window tab of the Configure Alarms dialog box
Message length (in characters), Version 2 or later firmware	1 – 256	1 – 144	0 – 80
Message length (in characters), earlier versions	1 – 72	1 – 70 single width 1 – 35 double width	0 – 50
Maximum number of lines displayable (per message)	limited by object size	3	Single-line window: 1 Multi-line window: 1 Panel window: 1 – 6
Maximum number of messages displayable at one time	1	1	Single-line window: 1 Multi-line window: up to 20, depending on text size Panel window: 1

Note: If you use a message length that is above the range listed in the “Message length, earlier versions” row, the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

The maximum message lengths described above are independent of the font and size of the message window. If the message is too long to fit in the message window, it will be truncated. The character “>”, indicating there is more text in the message, appears at the end of the displayed message if the message is truncated.

If the message is too long to fit on a single line, and the window can display more than one line for the message, the message will wrap to the next line after a break between words. Or, you can use the Newline code to specify where you want the line to break. See “Using Formatting Codes in Your Messages,” next.

Using Formatting Codes in Your Messages

Formatting codes give the PanelView terminal instructions on how to display your message.

For a Version 2 or later application, you can use these formatting codes in your messages:

This code:	Does this:
\n	Displays the text following the code on the next line.
\\n	Displays the text following the code without a carriage return. (See Example 1 on the next page.)

For Version 2 or later applications, the Newline code (\n) counts as one character for purposes of determining message length.

When a Version 2 or later application is downloaded to the PanelView terminal the Newline code is converted to a carriage return and line-feed sequence. For a pre-Version 2 application, the \n character sequence is not converted to a carriage return and line feed.

When a Version 2 or later application is uploaded from the PanelView terminal, the carriage return and line-feed is converted back to \n.

Note: For pre-Version 2 applications, \n is not converted into a new line. The \n appears as is, in a message format.

Note: For alarm messages, the Newline code works only in the Panel Alarm Message Window. In the other Alarm Message Windows and alarm screens the code appears as this symbol: ⊡

Version 2 or Later Applications

You can use formatting codes only for applications that are compatible with PanelView Version 2 or later. If you are creating a Version 2 or later application, and you want to use the character sequence \n, you must enter it as \\n. Otherwise PanelBuilder will interpret it as a carriage return and line feed. PanelBuilder strips out the extra backslash when creating the Version 2 or later .pvd file, so the characters in your message will appear as \n on the PanelView terminal.

The following examples illustrate how messages containing the \n character sequence appear in PanelBuilder and PanelView.

Example 1: Using the Newline Code in Version 2 and Later Applications

When you do this:	The message appears like this:
Type the code in PanelBuilder Version 2 or later	The conveyor has stopped.\nClear feeder and \\notify supervisor.
Display the message on the PanelView terminal Version 2 or later	The conveyor has stopped. Clear feeder and \notify supervisor.
Upload the application from PanelView into PanelBuilder Version 2 or later	The conveyor has stopped.\nClear feeder and \\notify supervisor.

Example 2: Using \n in Pre-Version 2 Applications

When you do this:	The message appears like this:
Type the backslash in PanelBuilder	Paper jam. Clear feeder\notify supervisor\wait for signal.
Display the message on the PanelView terminal	Paper jam. Clear feeder\notify supervisor\wait for signal.
Upload the application from PanelView into PanelBuilder pre-Version 2 or PanelBuilder Version 3	Paper jam. Clear feeder\notify supervisor\wait for signal.
Upload the application from PanelView into PanelBuilder Version 2.02	Paper jam. Clear feeder\\notify supervisor\wait for signal.

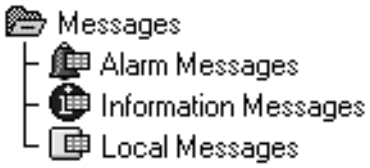
Note: Messages in pre-Version 2 applications remain unchanged.

Note: When a pre-Version 2 application is uploaded and the user changes the target firmware version to Version 2 or later, the messages will not be automatically converted or updated from \n to \\n.

Creating a New Message

You can create messages in both the form area and the spreadsheet.

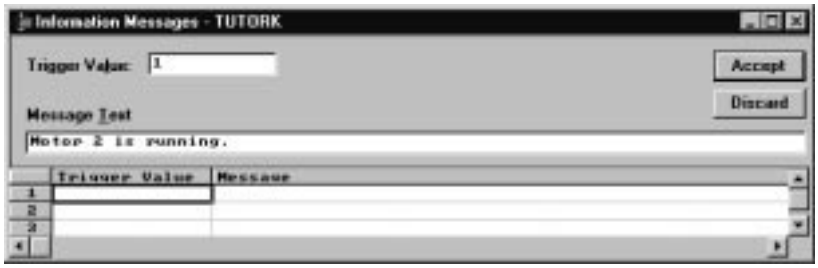
To create a message in the spreadsheet:



1. Open the message editor by choosing the appropriate editor in the Messages folder, or tool from the toolbar:
   - for alarm messages, choose
   - for information messages, choose
   - for local messages, choose

   The message spreadsheet appears by default. The following illustration shows the Information Messages spreadsheet.

   Note: For details on using the spreadsheet/form style editor, see Chapter 4, *Working with Editors*.



2. Position the mouse pointer in the first available Trigger Value field and click, or use the down arrow (↓) to move to the field.
3. Type a trigger value and press ENTER.

   Note: The trigger value cannot be 0. It must be unique for that editor.

4. Move to the Message field. Use TAB to move to the field, press the right arrow (→), or move the mouse pointer to the field and click.
5. Type the message text in the Message field next to the trigger value and press ENTER.

To edit a message in the spreadsheet:

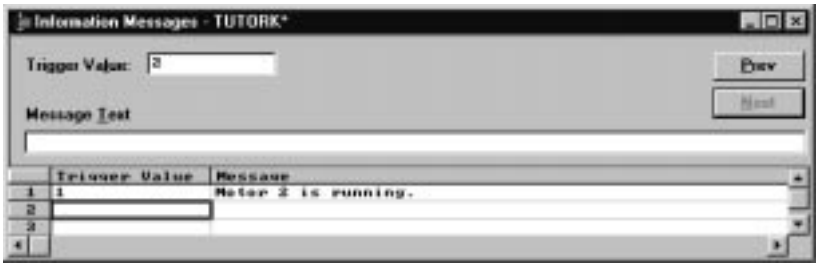
1. Open the spreadsheet.
2. Select the trigger value or message text you want to edit.
3. Double-click to activate the text insert bar, or simply begin typing.
4. Edit the message text or trigger value as you wish, and press ENTER when you're finished.

To create a message in the form:

1. Open the message editor by selecting the appropriate editor in the Messages folder or the tool from the toolbar.
2. Open the form, by checking Form in the View menu.
3. If there are existing messages, choose Next until you arrive at the first blank record.
4. Select the Trigger Value field and enter a unique trigger value.

   Note: If you do not enter a trigger value, the row number is the default trigger value.

5. Select the Message Text field and enter the message text.
6. Choose Accept.



7. Choose Next if you want to create another message.

To edit a message in the form:

1. Open the form.
2. Use the Prev or Next button to move to the message you want to edit.
3. Edit the message text or trigger value.
4. Choose Accept or press ENTER.

Adding Audio, Print, Display, Alarm Relay, and Message to PLC Options to Alarm Messages

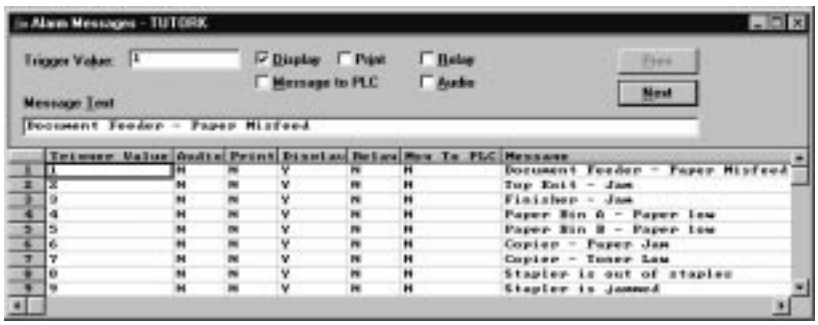
Creating and editing alarm messages is identical to the procedure described previously, except that you can enable one or more of these options for each alarm message:

- **Audio** sounds a continuous alarm when the alarm is triggered.
- **Print** sends the alarm message text to a printer when the alarm is triggered. The printer prints each alarm message with the time and date when the alarm occurred. The alarm message will be printed again with the time and date when the alarm is acknowledged.

  You can configure the printer's form feeds and line feeds through the PanelView terminal's Printer Setup screen. See the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual* for information about configuring the printer.
- **Display** sends the message to the Alarm Window. This option is enabled by default. If this option is disabled, the message will still appear in the Alarm History and Alarm Status screens, if these are enabled, but will not be displayed in the Alarm Window.
- **Relay** energizes the alarm relay when the alarm is triggered. This switches on whatever device is connected to the relay, such as a warning light or remote audio alarm.
- **Message to PLC** sends the alarm message to the PLC, where it can be retrieved by a third-party utility or another PanelView terminal. This option is disabled by default. To use this option you must assign the Alarm Message controls in the Global Controls tab of the Configure Alarms dialog box. See Chapter 11, *Configuring Alarms*, for more information.

  If you use this option, the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

The default for the Audio, Print, Relay, and Message to PLC options is "N" for "No". For Display, the default is "Y" for "Yes". You can change these options when you are creating a new message, or when editing an existing message.


Alarm Messages - TUTORK
Trigger Value:
Display
Print
Relay
Message to PLC
Audio
Prev
Next
Message Text
Document Feeder - Paper Misfeed
Trigger Value
Audio
Print
Display
Relay
How To PLC
Message
1 N N Y N N Document Feeder - Paper Misfeed
2 N N Y N N Top Exit - Jam
3 N N Y N N Finisher - Jam
4 N N Y N N Paper Bin A - Paper low
5 N N Y N N Paper Bin B - Paper low
6 N N Y N N Copier - Paper Jam
7 N N Y N N Copier - Toner Low
8 N N Y N N Stapler is out of staples
9 N N Y N N Stapler is jammed

To change an alarm message option in the spreadsheet:

- Select the cell of the option that you want to change.

  Double-click to change the N to a Y, or the Y to an N, or press TAB, SHIFT+TAB, or the arrow keys to move through the options. Press the BACKSPACE key twice to delete the existing option, and type Y or N to change the option as desired, then press ENTER.

To change an alarm message option in the form:

- Click in the field of the option that you want to change, or use TAB or ALT+the key for the underlined letter to move to the box, and press SPACEBAR.

Importing Messages

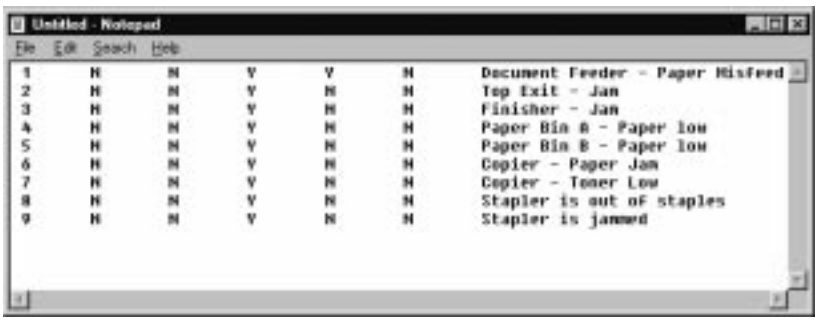
You can copy messages from other Windows spreadsheets, text editors, or word processors, and paste them into the message editors. Or, you can use the Import Message File command. Both methods for importing messages are described in this section.

Note: If you are importing messages into an existing message file, they are inserted above the currently-selected row in the Messages spreadsheet. Because each application can contain a maximum of 2000 messages, you will see the error message, “You have reached the maximum number of messages” if the messages you are importing increase the total number of messages to more than 2000.

Importing Messages Using the Clipboard

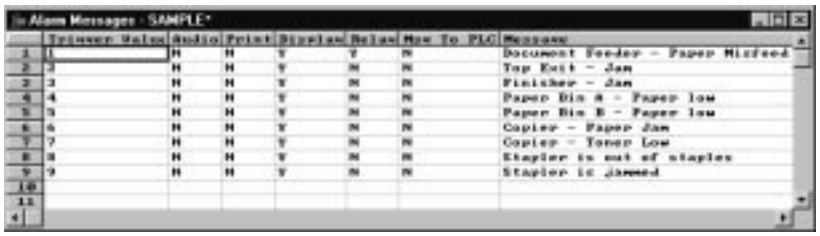
You can import messages by copying fields, rows, or columns of text from a text editor. The text entries are stored on the clipboard. You can then paste them into the message editor.

To copy and paste a single column, each column field must be on a separate line. To copy and paste one or more rows, each row must begin on a separate line. Each field in a row must be separated from the previous one by a tab. The following illustration shows an example of how the messages could appear.


Untitled - Notepad
File Edit Search Help
1 N N Y Y N Document Feeder - Paper Misfeed
2 N N Y N N Top Exit - Jam
3 N N Y N N Finisher - Jam
4 N N Y N N Paper Bin A - Paper low
5 N N Y N N Paper Bin B - Paper low
6 N N Y N N Copier - Paper Jam
7 N N Y N N Copier - Toner Low
8 N N Y N N Stapler is out of staples
9 N N Y N N Stapler is jammed

To copy messages into the spreadsheet using the clipboard:

1. In the text editor, select the message or messages to be copied.
2. Choose Copy.
3. In the message editor, select the field where you are going to begin pasting. To overwrite existing messages, select the rows you want to overwrite. To paste the messages into a larger or smaller area, select that area.
4. Choose Paste from the Edit menu, from the toolbar, or press CTRL+V.
5. The first message appears in the row whose cell you selected. Any other messages are pasted into the cells below. If you selected an area in the spreadsheet, the messages are pasted into that area.


Alarm Messages - SAMPLE*
Trigger Value
Audio
Print
Display
Relay
Msg To PLC
Message
1 N N Y Y N Document Feeder - Paper Misfeed
2 2 N N Y N N Top Exit - Jam
3 3 N N Y N N Finisher - Jam
4 4 N N Y N N Paper Bin A - Paper low
5 5 N N Y N N Paper Bin B - Paper low
6 6 N N Y N N Copier - Paper Jam
7 7 N N Y N N Copier - Toner Low
8 8 N N Y N N Stapler is out of staples
9 9 N N Y N N Stapler is Jammed
10
11

You can also export messages by copying them from the message editors and pasting them into other Windows text editors, as described in “Exporting Messages,” later in this chapter.

Importing Existing Message Files

You can also import existing message text, for example, from a text editor or old application files, and convert the text into alarm, information, or local messages by using the “Import Message File” command.

Guidelines for Importing Message Files

If you want to use the “Import Message File” command to import messages, follow these guidelines in preparing your message file:

- The trigger value can be any value from 1 to 2000. The values do not have to be created in any order, but no two messages can have the same trigger value.
- The message text must be surrounded by quotation marks (“...”).
- To include a quotation mark (”) in your message, preface it with a second quotation mark (”).
- PanelBuilder will ignore any spaces found outside the quotation marks.

- A character with an ASCII code below 32 is not valid and won't be imported. See the *ASCII Displays* chapter in the *PanelBuilder 1400e Configuration Software for Windows Reference Manual* for details of the ASCII character set. Also see Appendix B, *The Extended Character Set*, in this manual,
- Use the comment character (!) as the first character in the line of a comment. PanelBuilder will ignore lines that begin with !
- The number of characters in each message should be:
  - 80 characters or less for alarm messages
  - 144 characters or less for information messages
  - 256 characters or less for local messages
- When you name your file, use one of the following extensions:
  - .img for information messages
  - .lmg for local messages
  - .amg for alarm messages

Format of the Message File

Before you begin importing existing message files using the "Import Message File" option, ensure that the file is an ASCII text file. Your messages must be in this format:

```
FFF nnnn "ttt...ttt" CR
!this is a comment
```

FFF	=	alarm option (alarm messages only) A = audio P = print R = relay Include A, P, or R only if you want the option to be Yes. Display is automatically set to Yes, and Message to PLC is set to No, but you can edit these options once the file is imported.
nnnn	=	trigger value (any integer from 1 to 2,000) You can use 1 to 4 digits for the trigger value.
" "	=	start and end of message
ttt	=	message text 0 to 80 characters for alarm messages; 1 to 144 characters for information messages; 1 to 256 for local messages Messages that are longer than the maximum are still imported; however, they are detected as errors when the application is validated. You'll have to change them to the required length.

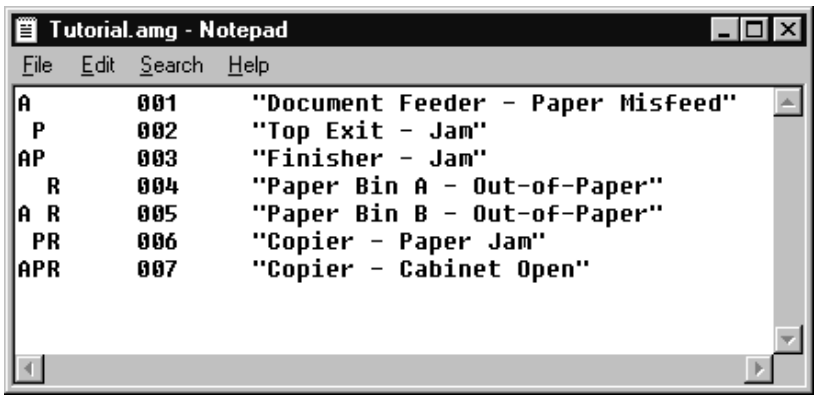
CR = Carriage return (record delimiter)

Press RETURN or ENTER on your keyboard.

! = comment character

Note: The message components—such as trigger values and message text—must be separated by spaces. Do not use .csv (Comma Separated Variable) files, as PanelBuilder does not yet support them for importing message text.

The following illustration shows a sample alarm message text file, created in Notepad in Microsoft Windows.


Tutorial.amg - Notepad
File Edit Search Help
A 001 "Document Feeder - Paper Misfeed"
P 002 "Top Exit - Jam"
AP 003 "Finisher - Jam"
R 004 "Paper Bin A - Out-of-Paper"
A R 005 "Paper Bin B - Out-of-Paper"
PR 006 "Copier - Paper Jam"
APR 007 "Copier - Cabinet Open"

To import messages using the Import Message File command:

1. Choose Import Message File from the File menu.

   A dialog box appears. If it doesn't show the file you want to import, you may need to change the drive and directory.
2. Select the file.
3. Choose OK.

   If there are no errors in the format of the message file, it appears in your message list above any existing messages.
4. Choose Update Messages from the File menu or from the toolbar to save the messages.

Note: The messages you are importing are added to the message database at the current cursor position. They are automatically re-sorted by trigger-value when you next open the Message Editor. You can save a maximum of 2000 alarm messages. If the messages you are importing increase the number of messages in the message database beyond 2000, you will see an error message, and only the first 2000 messages are saved.

Exporting Messages

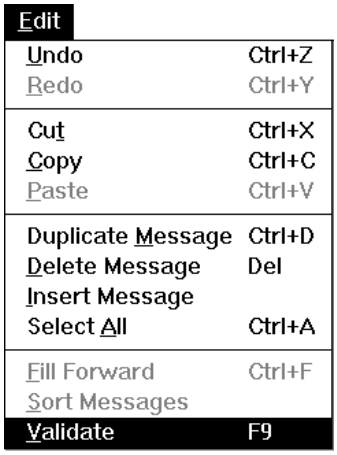
To export messages, you must use the clipboard. You can copy the messages, trigger values, and alarm notification options from the message editor, and paste them into the text editor of your choice.

To export messages using the clipboard:

1. Select the area of the spreadsheet you want to copy.
2. Choose Copy from the Edit menu, from the toolbar, or press CTRL+C.
3. Open the text editor to which you want to copy the messages.
4. Paste the messages into the text editor.

Validating Messages

You can validate your messages anytime, in any of the views, to ensure that you have followed all the guidelines for creating and editing messages.



To validate your messages:

1. Choose Validate in the Edit menu.

   A Validation Results window appears. If your message list contains errors, such as duplicate trigger values, you'll see them listed in the window. If there are no errors in the message list, a check mark appears.

Validation Results for Alarm Messages - Untitled1
Total number of errors: 3
Close
Update
Tag Usage Viewer
Print...
Print Setup...
Help
Alarm Messages (a total of 3 errors were found)
1. Row 2: The trigger value cannot be zero.
2. Row 4: Alarm message text can be blank only if audio, print, relay or messag
3. Row 3: This trigger value is currently in use by another message. Duplicate

2. If there are errors listed, go back to the screen or the message editor and correct the errors.

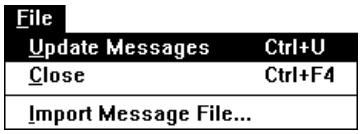
Tip: You can leave the Validation Results window open on your desktop while you're making the necessary changes to the message list. You may want to resize the window and move it slightly off-screen. When you've made a correction, choose Update in the Validation Results window. This will check the changes you've made. You can also print the Validation Results window.

Updating Messages

While you are creating messages, you should update (save) them from time to time. When you close your message editor, you are prompted to save your messages.

PanelBuilder automatically validates your messages before updating them.

To update messages:



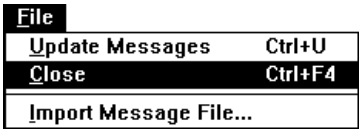
1. Choose Update Messages from the File menu, from the toolbar, or press CTRL+U.

   If there are no errors, a message informs you that the messages have been updated. If there is an error in your messages, a dialog box informs you of the error.
2. Follow the steps of the previous section for validating your messages.
3. Update your messages again. A message informs you when the messages are updated.

Your messages must be error-free before you can update them or close the message editor.

Important: If you close your application without saving it, the changes you made to your messages will be discarded, even though you updated them. Save your application frequently. For details on saving the application, see Chapter 3, *Working with Applications*.

Closing the Messages Editor



To close the editor:

1. Choose Close from the File menu, or double-click in the control box of the editor.

   If you made any changes and haven't updated them, a message appears, prompting you to save your changes.

2. Choose Yes or No, depending on whether you want to update the changes.

   If you choose Yes, PanelBuilder automatically checks your messages for errors. If there are any errors, PanelBuilder informs you of these. Follow the steps outlined in the previous section, and correct the messages.

   If no errors exist, the messages are updated and the editor closes.

   Important: If you choose No, PanelBuilder closes the editor without updating the messages. You'll lose any changes you made.

   If you change your mind and don't want to close the editor, choose Cancel.

   Important: Updating your messages stores the changes only temporarily. If you exit the application without saving it, these updates will be lost. To ensure that you save all changes, save your application frequently. For details on saving the application, see Chapter 3, *Working with Applications.*

Configuring Alarms

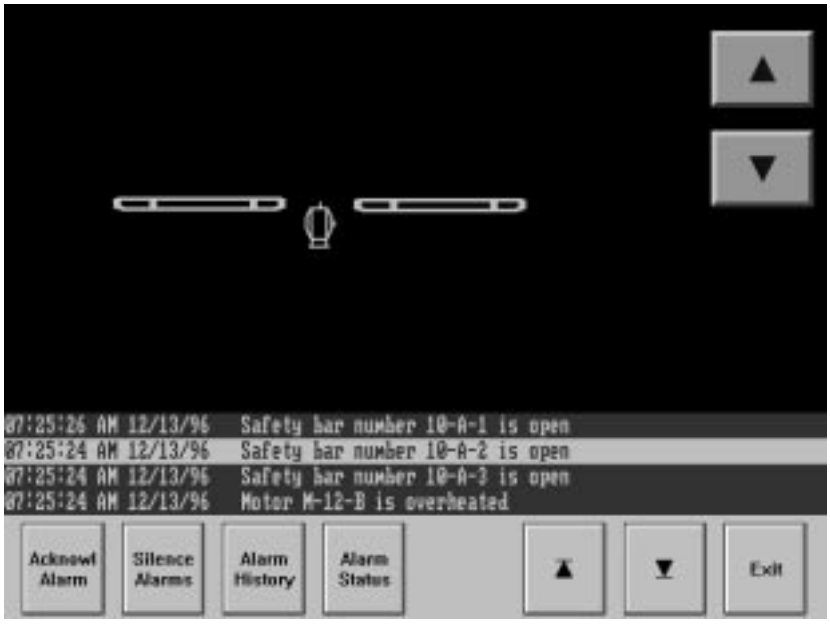
This chapter describes these topics:

- the interaction between the terminal and the PLC program that generates alarms
- the global controls used to report alarms and how to configure them
- the Alarm Message Window and how to configure it
- the Alarm Status screen and how to configure it
- the Alarm History screen and how to configure it

What Are Alarms?

Alarms indicate to an operator that a ‘priority condition’ has occurred which requires immediate attention. An alarm can be indicated in a number of ways including:

- Displaying alarm message text on the PanelView terminal’s screen
- Triggering PanelView’s alarm relay to energize
- Triggering PanelView’s audio beeper to sound
- Sending alarm message text to a printer
- Sending alarm message text to the PLC for use by a supervisory computer.


07:25:26 AM 12/13/96 Safety bar number 10-A-1 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-2 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-3 is open
07:25:24 AM 12/13/96 Motor M-12-B is overheated
Acknowl Alarm
Silence Alarms
Alarm History
Alarm Status
Exit

ATTENTION: Do not rely on your terminal as a primary warning device, or as an indicator of a critical or dangerous situation. “Critical” means any situation that could result in physical injury, product damage, or significant process down time.

An operator’s response to a critical or dangerous situation must never depend on software or solid state electronics. All critical alarm indications and any critical host controller responses must employ redundant and hard-wired or mechanical interlocks.

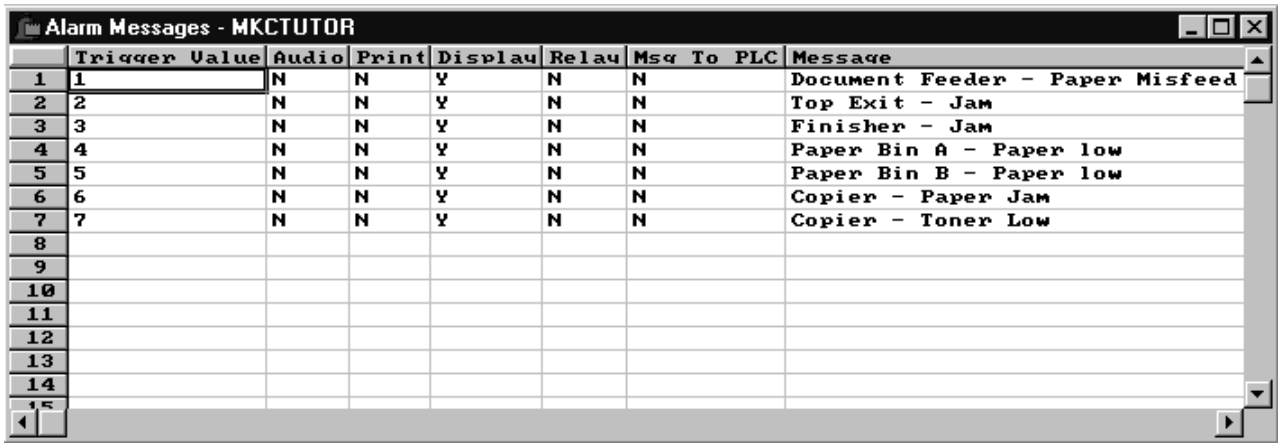
How Are Alarms Triggered?

Before the PanelView terminal can report alarms, you must define the alarm conditions in the application file and program the PLC to trigger the alarms.

In PanelBuilder you create a list of alarm messages. For each message you specify text and a unique trigger value. The PLC must be programmed to monitor alarm conditions and, when an alarm is to be generated, send the trigger value that corresponds to that alarm to the PanelView terminal. An alarm trigger tag or expression, defined in PanelBuilder, is required to send the trigger value from the PLC to the PanelView terminal. Up to 50 different tags or expressions can be configured to trigger alarms from multiple locations.

Alarm Indication Options

When configuring Alarm Messages in the Alarm Messages editor (see Chapter 10), each message can be configured to sound PanelView’s **Audio** alarm, **Print** the message on a printer, **Display** the alarm in the Alarm Message Window, energize PanelView’s built-in **Relay**, or send the triggered alarm **Message to the PLC** where it can be sent to other devices such as a supervisory computer.



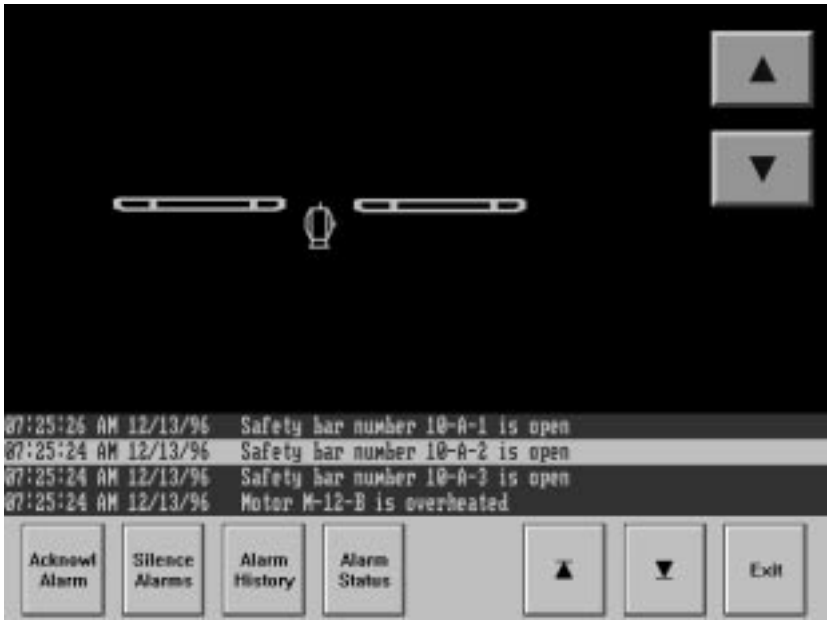
	Trigger Value	Audio	Print	Displau	Relau	Msg To PLC	Message
1	1	N	N	Y	N	N	Document Feeder - Paper Misfeed
2	2	N	N	Y	N	N	Top Exit - Jam
3	3	N	N	Y	N	N	Finisher - Jam
4	4	N	N	Y	N	N	Paper Bin A - Paper low
5	5	N	N	Y	N	N	Paper Bin B - Paper low
6	6	N	N	Y	N	N	Copier - Paper Jam
7	7	N	N	Y	N	N	Copier - Toner Low
8							
9							
10							
11							
12							
13							
14							
15							

How Are Alarms Displayed?

Alarms can be displayed in three ways:

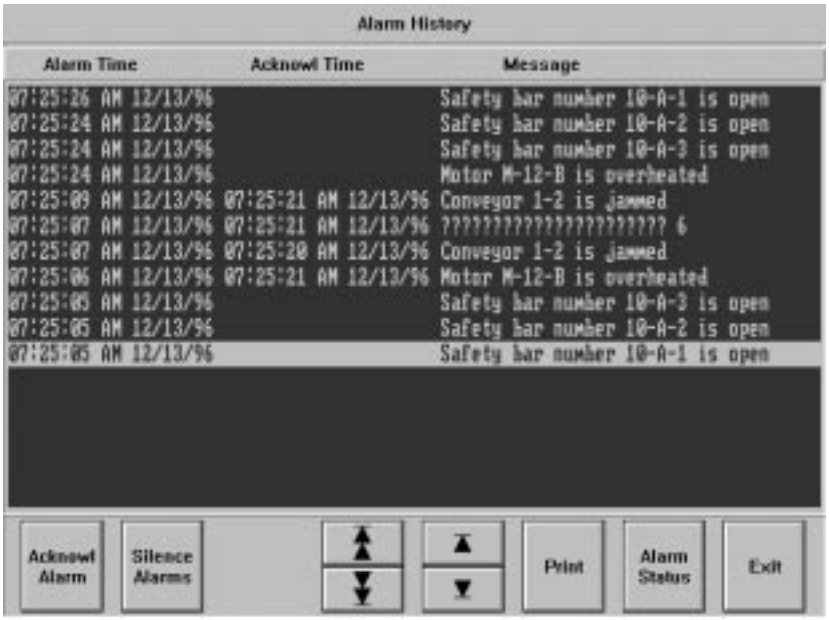
- Message Window
- Alarm History screen
- Alarm Status screen

Alarm Message Windows display unacknowledged alarms. They pop up over any user-configured screen, and the information they display is not retained across power cycles of the PanelView terminal. The user can silence and acknowledge alarms from the multi–line Alarm Message Window.

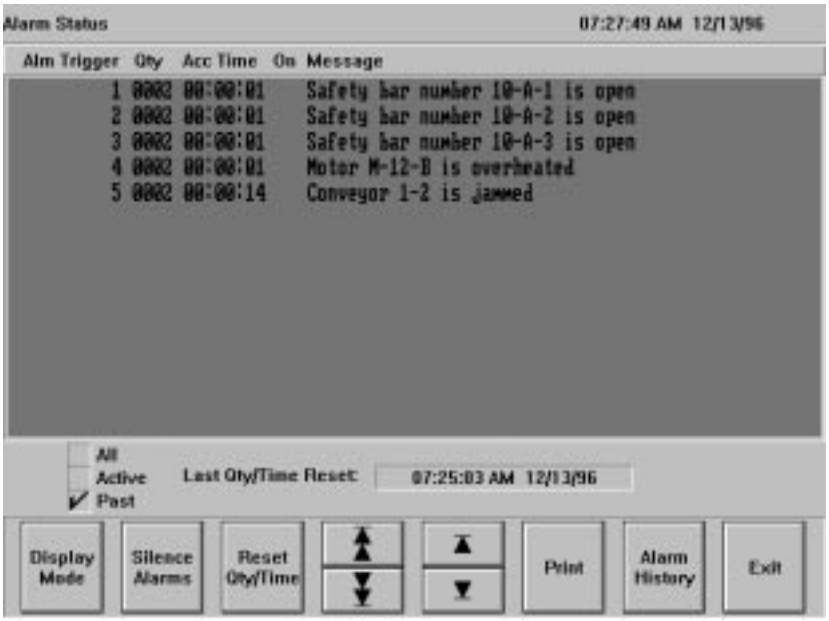

07:25:26 AM 12/13/96 Safety bar number 10-A-1 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-2 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-3 is open
07:25:24 AM 12/13/96 Motor M-12-B is overheated
Acknowl Alarm
Silence Alarms
Alarm History
Alarm Status
Exit

The **Alarm History screen** provides a record of the last 128 alarms that have occurred and is maintained between power cycles on the PanelView terminal. The user can silence and acknowledge alarms from this screen as well.

You can upload the alarm history file from the PanelView terminal to a development computer, using the PanelView 1200/1400e Transfer Utility. For more information, see the *PanelView 1400e Transfer Utility User Manual.*


Alarm History
Alarm Time
Acknowl Time
Message
07:25:26 AM 12/13/96 Safety bar number 10-A-1 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-2 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-3 is open
07:25:24 AM 12/13/96 Motor M-12-B is overheated
07:25:09 AM 12/13/96 07:25:21 AM 12/13/96 Conveyor 1-2 is jammed
07:25:07 AM 12/13/96 07:25:21 AM 12/13/96 ?????????????????????? 6
07:25:07 AM 12/13/96 07:25:20 AM 12/13/96 Conveyor 1-2 is jammed
07:25:06 AM 12/13/96 07:25:21 AM 12/13/96 Motor M-12-B is overheated
07:25:05 AM 12/13/96 Safety bar number 10-A-3 is open
07:25:05 AM 12/13/96 Safety bar number 10-A-2 is open
07:25:05 AM 12/13/96 Safety bar number 10-A-1 is open
Acknowl Alarm
Silence Alarms
Print
Alarm Status
Exit

The **Alarm Status screen** provides a list of all the alarm messages in the application with an indication of whether they are active or inactive. The information displayed on the Alarm Status screen is retained across power cycles of the PanelView terminal.



Detailed information about configuring each of these methods of displaying alarms is provided later in this chapter.

How Are Alarms Acknowledged?

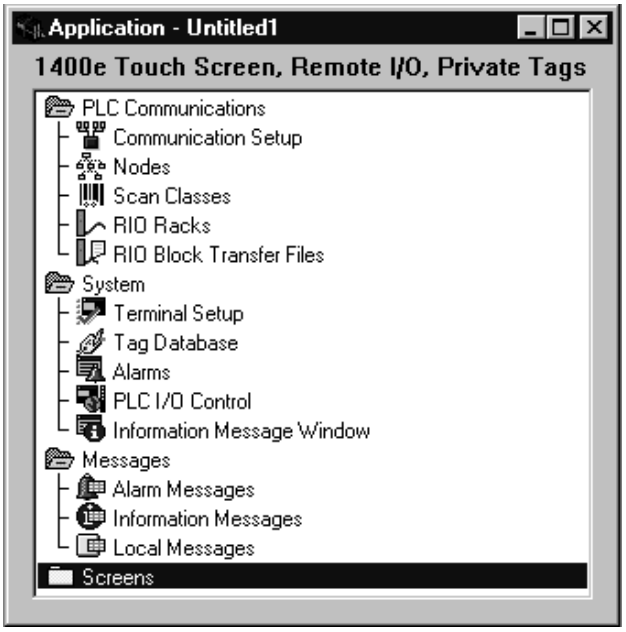
Alarms can be acknowledged by an operator pressing the Acknowledge button provided in the Multi-line Alarm Message Window or in the Alarm History Screen. They can also be remotely acknowledged by the PLC. See details later in this chapter on configuring the different acknowledge options.

When an alarm is acknowledged, the terminal does the following depending on how Alarms are configured:

- removes the alarm from the Alarm Message Window
- records the acknowledged time and date in the Alarm History screen
- silences the alarm beeper
- de-energizes the alarm relay
- prints the acknowledged alarm with its acknowledged time and date stamp
- sends the trigger value of the acknowledged alarm to the PLC

Configuring Alarms

You use three editors to configure alarms in PanelBuilder:



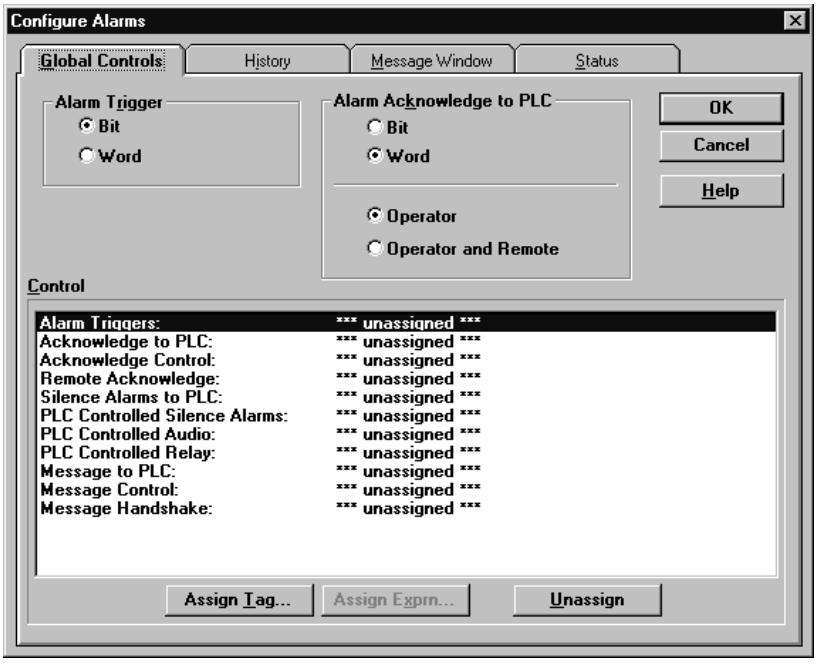
Use the **System/Terminal Setup** editor to set up Remote Alarm Timing parameters. See Chapter 12 for more details.

Use the **System/Alarms** editor to configure the Alarm Global Controls, the Alarm History screen, the Alarm Message Window and the Alarm Status screen.

Use the **Messages/Alarm Messages** editor to define all the alarm messages, their corresponding trigger values and Audio, Print, Display, Relay, and Message to PLC indication options. See Chapter 10 for more details.

Configuring Alarms' Global Controls

Select the System/Alarms editor to enter the Configure Alarms dialog box. Select the Global Controls tab.



Alarm Trigger (Bit or Word?)

Before deciding whether to use **Bit** or **Word** triggers, careful planning is essential. Your choice here depends on your application and how you plan to implement alarms. The following questions should be answered:

1. How many alarms need to be generated?
2. Do I need to monitor multiple alarms simultaneously or am I only concerned about the first alarm that occurs and want to ignore all subsequent alarms (because they probably occur as a result of the "domino effect" of the first alarm)?
3. Are alarms coming from multiple PLCs (DH+ or ControlNet)?
4. Do I need to perform arithmetic or logical operations to determine which alarm to display (for example, threshold alarming)?
5. Do I have any addressing space constraints?

In general, if monitoring of simultaneous alarms is required it is best to use Bit triggered alarms.

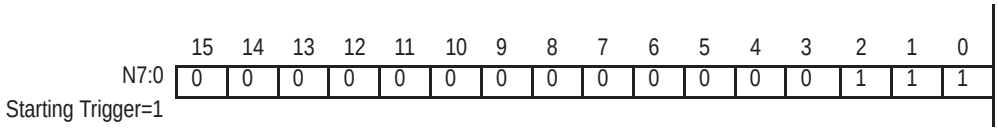
If monitoring of simultaneous alarms is not required, Word triggered alarms will generally save address space (and therefore update time) because one word can generate up to 2000 alarms (one at a time). Word trigger is also a good choice for monitoring threshold conditions because expressions can be assigned to Word triggered alarms. This can offload logic required in the PLC because PanelView does the calculations. Alarms can be generated from up to 50 different sources (on one or more PLCs) since up to 50 tags or expressions can be assigned.

Bit triggered alarms allow the PanelView terminal to capture simultaneous alarms within the same tag reference (you cannot define expressions for Bit triggered alarms). That is, more than one alarm within a tag can be triggered at a time. Here's how it works:

Example 1: Bit Triggered Alarms

Assume the following alarm messages and Alarm Trigger block tag starting at N7:0 are defined in the application. The tag's starting trigger value is 1.

Trigger Value	Message Text
1	Message 1
2	Message 2
3	Message 3
4	Message 4
5	Message 5
6	Message 6
7	Message 7



Bits 0, 1, and 2 on would trigger alarms "Message 1," "Message 2," and "Message 3"

Word triggered alarms can only capture one alarm at a time in a single tag (or expression). In the example above for Bit trigger, assume the same alarm messages are defined and the tag is defined as an analog tag. In this case only the alarm with trigger value 7 (Binary 7 = bits 0,1,2 On) would be displayed (“Message 7”).

Refer to the Control section below for more information on Bit and Word triggered alarms and for details on using multiple alarm tags or expressions.

Alarm Acknowledge to PLC

Depending on the options configured for an alarm, once it is triggered, it will continue to be displayed, beep, or energize the relay until it is acknowledged. This requires the operator to be aware of the alarm and take action before the alarm is removed.

An alarm can be acknowledged by either the operator or the PLC. The operator can acknowledge alarms from either the Multi-line Alarm Window or from the Alarm History screen. See “Using the Alarm Message Window in the Terminal,” later in this chapter, for more information.

The **Alarm Acknowledge to PLC** option allows the PanelView terminal to notify the PLC when an alarm acknowledge occurs (sets the “Acknowledge Control” bit) and to send the trigger value of the acknowledged alarm to the PLC (in the address defined for “Acknowledge to PLC”).

The **Bit** or **Word** selection here refers to the type of addressing used to send the trigger value of the acknowledged alarm to the PLC. (See the example above in the “Alarm Trigger (Bit or Word?)” section for the difference between Bit and Word). Your selection here is independent of your selection for the Alarm Trigger type. This selection is only required if you want to use the “Alarm Acknowledge to PLC” option. If not, it can be ignored and you can use the default selection.

In general, **Bit** Acknowledge to PLC will give you more flexibility in using the acknowledged trigger value in the PLC. For example, you can use the individual bits in the ladder logic to clear the alarm condition. (See the example in “Acknowledge To PLC and Acknowledge Control,” later in this chapter, for sample PLC logic required to do this.)

Use **Word** Alarm Acknowledge to PLC if you are short on address space or are concerned about scan time performance. It will take less address space and less time to transfer one word of data than to transfer several blocks of data.

The **Operator** or **Operator and Remote** selection defines when to activate the "Alarm Acknowledge to PLC" option. This selection is only required if you want to use that option. If not, it can be ignored.

If you choose this option:	The "Acknowledge Control" will be set and the acknowledged alarm trigger value will be sent to the PLC when:
Operator	The operator presses the acknowledge button on the Multi-line Alarm Message Window or on the Alarm History screen.
Operator and Remote	The operator presses the acknowledge button on the Multi-line Alarm Message Window or on the Alarm History screen **and/or** when an alarm is acknowledged remotely by the PLC.

Note: If you set the Acknowledge to PLC option to Operator and Remote, the application will be compatible with PanelBuilder Version 2 or later. It will not be compatible with earlier versions.

Control

You must assign the "Alarm Triggers" before any alarms can be reported to the PanelView terminal. The remaining Global Controls are optional.

Alarm Triggers

Alarm Triggers are tags or expressions used by the PanelView terminal to monitor alarm conditions generated at the PLC. Assignment of Alarm Trigger(s) is required for alarms to function. Up to 50 different alarm tags or expressions can be assigned to monitor alarms from multiple sources. The tag type or expression allowed depends on your selection for Alarm Trigger type as follows. Invalid tag types will not be reported until validation.

If you selected this Alarm Trigger type:	Expressions allowed?	Max. no. of alarm tags / expressions:	You must use this tag type:	Valid "Starting Trigger":
Bit	No	50 (tags only)	Block	1 to 65,535
Word	Yes	50 (tags or expressions)	Digital and/or Analog	N/A

Note: For DH+ and ControlNet applications, if you assign block tags to the Alarm Triggers control or Acknowledge to PLC control, the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Note: For all applications, if you assign expressions to any control, the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Note: For all applications, if you assign more than one Alarm Trigger tag the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Bit Trigger Operation

For a **block tag**, each bit in the tag that makes a 0 to 1 transition will trigger an alarm whose trigger value is equal to the tag's bit position added to the starting trigger value.

If more than one alarm is triggered at the same time, the alarms are logged to the Alarm History file in ascending numerical order according to their trigger value. If the alarms span multiple tags, the alarms will be logged according to the order in which they were configured in the Configure Alarms dialog box. Simultaneous alarms will be stamped with the same time and date, regardless of whether they were triggered from a single tag or multiple tags.

Starting Trigger

Each block tag must be assigned a starting trigger value from 1 to 65,535 (you can assign a maximum of 2000 alarms). The starting trigger is the trigger value of the alarm that is to be triggered by the first bit of the block tag. For example, if a block tag is configured for 10 bits and a Starting Trigger value of 1, it can be used to trigger alarms with trigger values 1 to 10. If that same block tag has a Starting Trigger value of 3, it can be used to trigger alarms with trigger values 3 to 12.

Important: For Bit triggered alarms, each message must be uniquely triggered by a single block tag. That is, different block tags cannot trigger the same alarm.

Example 2: Using Multiple Alarm Tags for Bit Trigger Operation on Remote I/O

Assume the following alarm messages and block tags are defined in the application.

Trigger Value	Message Text
1	Message 1
2	Message 2
3	Message 3
4	Message 4
5	Message 5
6	Message 6
7	Message 7
8	Message 8

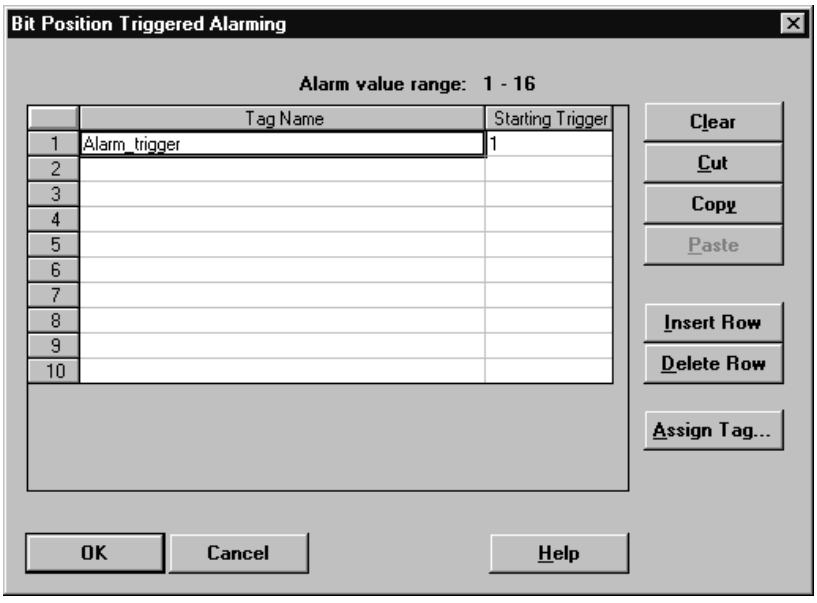
Trigger Value	Message Text
9	Message 9
10	Message 10
11	Message 11
12	Message 12

Block Tag Name	Address	Length	Starting Trigger	Messages Triggered
Alarm_tag1	O:010/00	4	1	Messages 1 to 4
Alarm_tag2	O:010/04	5	5	Messages 5 to 9
Alarm_tag3	O:011/00	3	10	Messages 10 to 12

If the bits indicated in the following data table are turned On, Messages 1, 2, 6, 8, and 12 will be triggered.

Block Tags	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
O:010	0	0	0	0	0	0	0	0	1	0	1	0	0	0	1	1
O:011	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

To assign tags to Bit triggered Alarm Triggers:



1. In the Configure Alarms dialog box, select the Global Controls tab.

2. Go to Control and double-click Alarm Triggers, or click Alarm Triggers and choose Assign Tag.

   The Bit Position Triggered Alarming dialog box appears with a table of tags and their starting triggers. Click on a row to enter the tag.

3. Choose the Assign Tag button and the Tag Browser dialog box appears. From this dialog box you can choose an existing tag (by double-clicking it) or create a new one (by choosing New Tag).

   If you choose New Tag, the Tag Editor dialog box appears. For details on using the Tag editor, see "Other Methods of Creating Tags" in Chapter 6, *Defining Tags.*

4. In the Bit Position Triggered Alarming dialog box, assign a Starting Trigger value from 1 to 65,535 for each tag. See "Starting Trigger" above, for more details.

5. To assign multiple tags (up to 50), click on the next available row and enter the new tag, repeating steps 3 and 4.

   If additional rows are needed, choose Insert Row. To delete a row, choose Delete Row.

6. Select OK to return to the Global Controls dialog box.

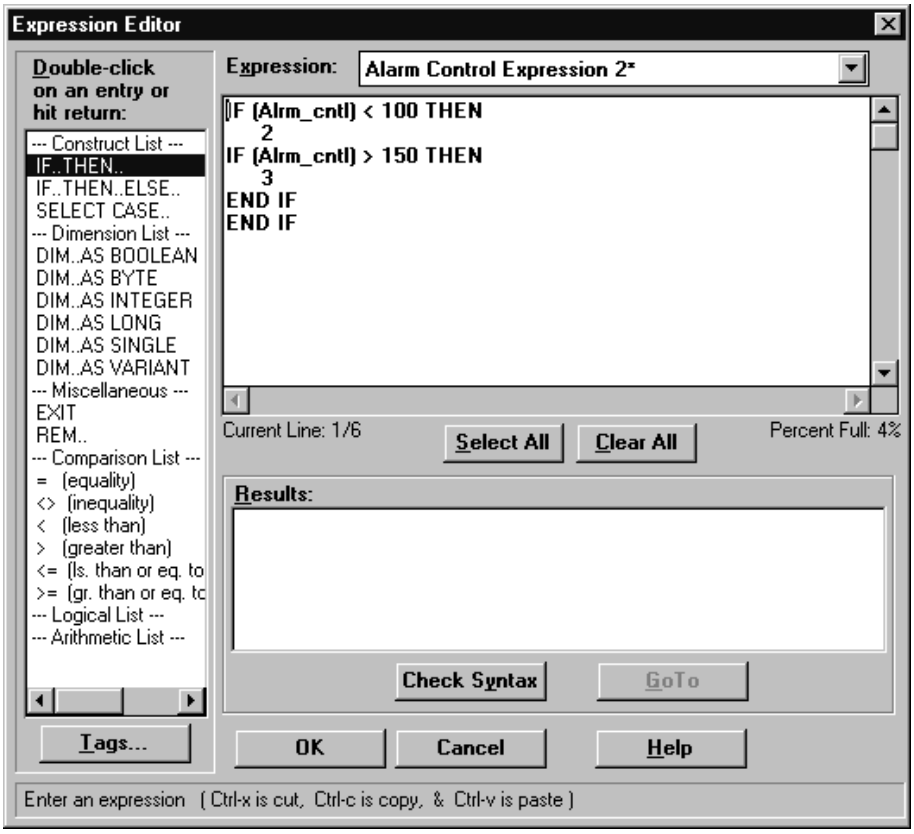
Word Trigger Operation

For a **digital tag**, when the tag value changes from 0 to 1, the alarm is triggered.

For an **analog tag**, any non-zero value will trigger an alarm with the corresponding trigger value.

When an **expression** is used, it will be evaluated any time a tag in the expression changes. Any non-zero expression result will trigger the alarm with the corresponding trigger value. Expressions allow the PanelView terminal to perform calculations on control values and determine internally whether an alarm state has occurred, and then trigger the alarm. This reduces ladder logic required in the PLC.

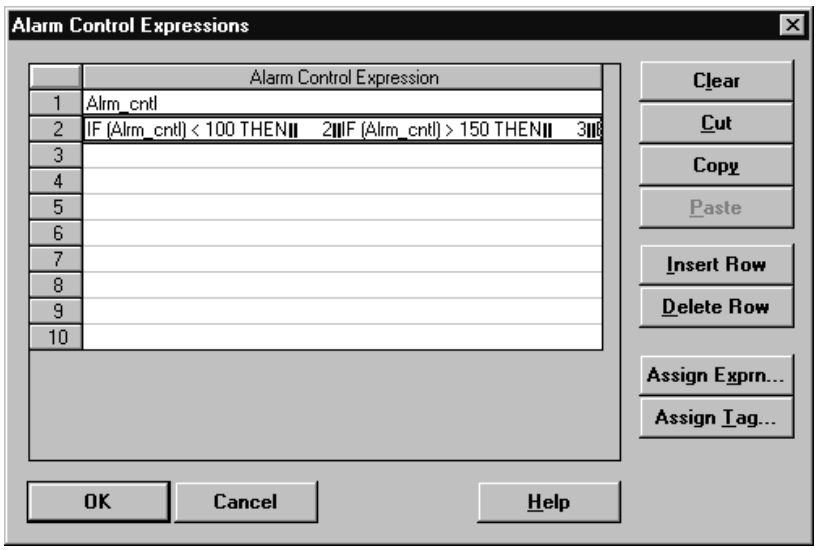
In the following expression example, the alarm tag "Alrm_cntl" is used to monitor temperature. If the temperature falls below 100°F, then the alarm message with a trigger value of 2 will be triggered. The alarm message with a trigger value of 3 will be triggered if the temperature rises above 150°F.



Using Multiple Alarm Tags for Word Trigger Operation

If multiple alarm tags or expressions are defined and more than one alarm is triggered at the same time, the alarms are logged to the Alarm History file according to the order in which they were configured in the Alarm Messages dialog box. However, they will be stamped with the same time and date.

Any Word triggered alarm can be triggered by any of the alarm tags or expressions. That is, the same alarm can be triggered by different tags or expressions. If the same alarm is triggered by multiple tags or expressions in the same PLC scan, multiple instances of that alarm will be indicated.



To assign tags or expressions to Word triggered Alarm Triggers:

1. In the Configure Alarms dialog box, select the Global Controls tab.
2. Double click Alarm Triggers, or click Alarm Triggers and choose Assign Exprn (Expression). The Alarm Control Expressions dialog box appears.

   Note: Choose Assign Expression even if you want to enter a tag.

3. Click on a row to enter the tag or expression.

   To assign a tag, choose Assign Tag.

   The Tag Browser dialog box appears. From this dialog box you can choose an existing tag (by double-clicking it) or create a new one (by choosing New Tag). If you choose New Tag, the Tag Editor dialog box appears. For details on using the Tag editor, see "Other Methods of Creating Tags" in Chapter 6, *Defining Tags*.

   To assign an expression, choose Assign Exprn (Expression). The Expression editor appears.

   For details on using the Expression editor see Chapter 7, *Creating Expressions.*

4. **To assign multiple tags or expressions** (up to 50), click on the next available row and enter the new Tag or Expression, repeating step 3.

   If additional rows are needed, choose Insert Row. To delete a row, choose Delete Row.

5. Select OK to return to the Global Controls dialog box.

Acknowledge to PLC and Acknowledge Control

The **Acknowledge to PLC** and **Acknowledge Control** controls are used together to inform the PLC when an alarm has been acknowledged and to send the acknowledged alarm's trigger value to the PLC. These controls are optional.

When these controls are activated depends on your choice for the Operator versus Operator and Remote selection (refer to the earlier section Alarm Acknowledge to PLC for more details). To enable the Acknowledge to PLC option, you must assign a tag to that control. The tag allowed for the Acknowledge to PLC control depends on your selection for Alarm Acknowledge to PLC type (Bit or Word). Invalid tag types will not be reported until validation. See "Alarm Acknowledge to PLC," earlier in this chapter, for more information on making these selections.

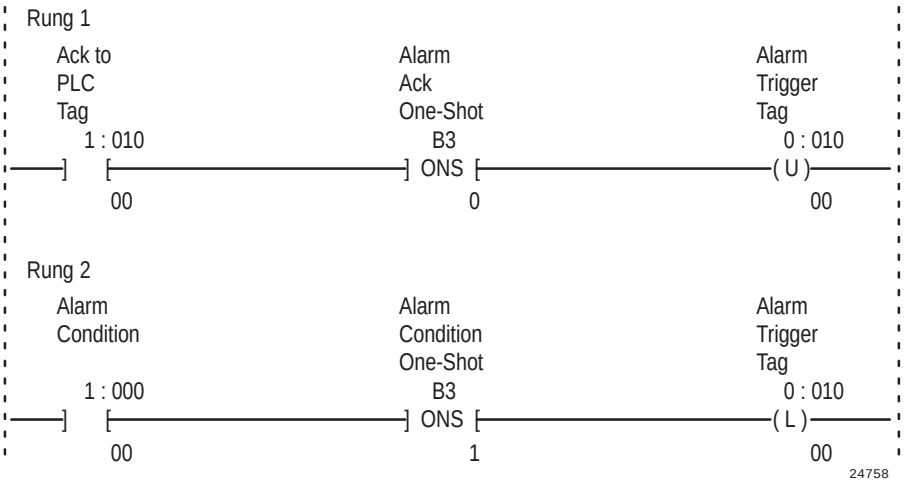
If you selected this Alarm Acknowledge to PLC type:	Expressions allowed?	Max. no. of Acknowledge to PLC tags:	You must use this tag type for the Acknowledge to PLC control:	Valid "Starting Trigger":
Bit	No	50	Block	1 to 65,535
Word	No	1	Digital or Analog	N/A

If the Acknowledge to PLC control tag type is this:	It does this when the alarm is acknowledged:
Block	Only the bit corresponding to the acknowledged alarm's trigger value will be set. For example, if an alarm with a trigger value of 14 is acknowledged and the starting trigger is set to 1, then the fourteenth bit in that block tag will be set to 1. The bit will remain set to 1 until that alarm is triggered again, when the bit will be reset to 0.
Digital	The tag will be set to 1.
Analog	The terminal will write the trigger value of the acknowledged alarm to the tag.

Note: If you press the Ack button on the Alarm History screen when no alarms are waiting to be acknowledged, all bits in the Acknowledge to PLC control tag are reset to 0. This lets you reset the digital and analog tags.

You can use the Acknowledge to PLC control tag to clear alarms as follows:

Example 3: Logic for Clearing Alarms when Using Bit Acknowledge to PLC Type with Block Tags



Comments:

Rung 1: When the Acknowledged alarm is received by the PLC, the alarm is cleared.

Rung 2: The alarm trigger value is sent to the PanelView terminal on a false-to-true transition of an alarm condition.

Note: The alarm condition must make another false to true transition to trigger another alarm message in the PanelView terminal. The Ack Control Tag ONS (one shot) instruction in the first rung is used because the Ack to PLC tag remains on until that same alarm is triggered again. It is important to place the unlatch rung before the latch rung to ensure proper alarm message triggering if both the alarm acknowledgment and new alarm condition occur in the same PLC program scan.

Using Multiple Acknowledge to PLC Tags for Bit Trigger Operation

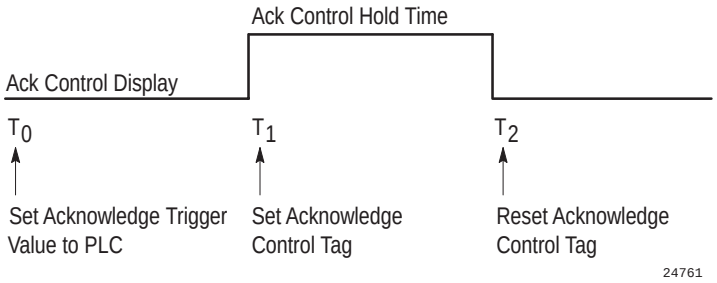
If you use the Bit Acknowledge to PLC type, you can assign up to 50 block tags to this control. Each tag must be assigned a starting trigger value from 1 to 65,535. The PanelView terminal uses the starting triggers to determine which block tag to send the alarm acknowledgment to, and sets the appropriate bit position of the tag to 1 where:

bit position = alarm trigger value – starting trigger

See the previous section, “Alarm Triggers,” for more information on Bit trigger and starting trigger operation.

To enable the Acknowledge Control, assign a digital or analog tag. When an alarm is acknowledged, and after the trigger value is written to the Acknowledge to PLC tag, the terminal sets the Acknowledge Control tag to 1 after the Alarm Control Delay interval. The Acknowledge Control tag is set to 1 for as long as the Acknowledged button is being pressed (for operator acknowledge), or for the Remote Alarm Ack Control Hold Time (for remote acknowledge), whichever is longer. After the Remote Alarm Ack Control Hold Time expires, the tag will be reset to 0.

Timing Diagram:



Note: The Alarm Control Delay time and Ack Control Hold Time are configured in the Timing Parameters tab of the Configure Terminal Setup dialog box. See “Configuring Timing Parameters” in Chapter 12, *Configuring Terminal Setup Options*, for more information about these parameters.

The following table shows the valid entries and default settings for the remote alarm timing parameters.

For this timing parameter:	Valid choices are these:		The default setting is this:
Operation Hold Time	50 msec 500 msec 1 second 3 seconds 5 seconds	250 msec 750 msec 2 seconds 4 seconds	3 seconds
Acknowledge Control Hold Time	50 msec 500 msec 1 second 3 seconds 5 seconds	250 msec 750 msec 2 seconds 4 seconds	250 msec
Alarm Control Delay	200 msec 600 msec 1 second 3 seconds 5 seconds	400 msec 800 msec 2 seconds 4 seconds	200 msec

If the Acknowledge Control tag is not assigned (disabled), the next acknowledged alarm trigger value will be written to the PLC immediately. It will not wait for the Alarm Control Delay. It is therefore recommended that you use the Acknowledge Control tag to control the flow of alarms and acknowledgments.

Only one alarm acknowledgment can be processed at a time; alarm acknowledgments are not queued. An acknowledge request is ignored and discarded (the alarm will not be removed from the Multi-line Alarm Window and the acknowledged time will not be recorded in the Alarm History screen) if it arrives when the terminal is:

- sending a previous acknowledge to the PLC or
- waiting for the Alarm Control Delay time or
- waiting for the Ack Control Hold Time.

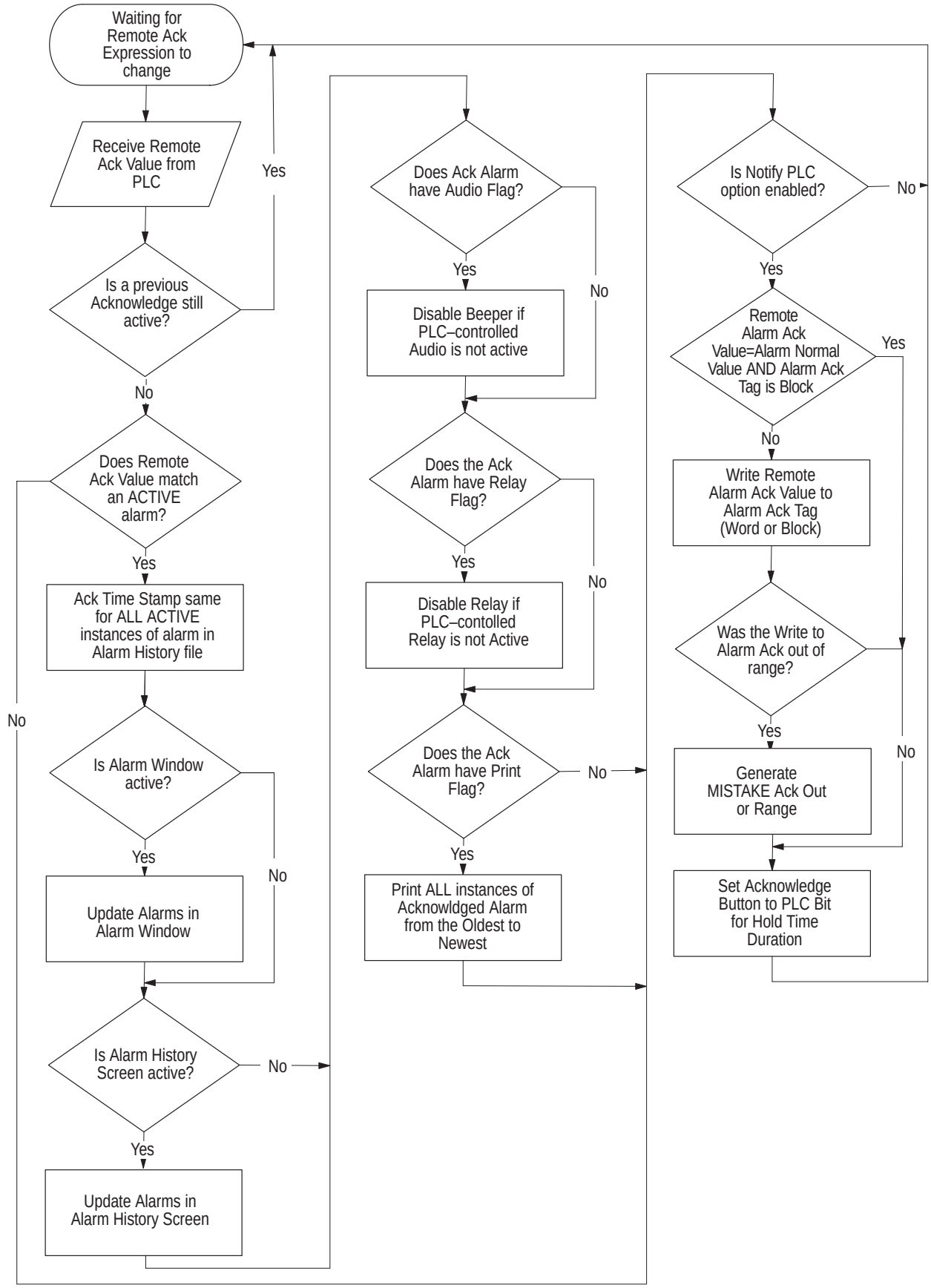
Remote Acknowledge

With **Remote Acknowledge** the PLC is used to acknowledge the alarm, rather than the operator acknowledging it from the terminal. This is useful for background alarming when there is no acknowledge button available, when using the Single-line Alarm Message Window or Alarm Message Panel, or when the operator is acknowledging the alarm from a remote location.

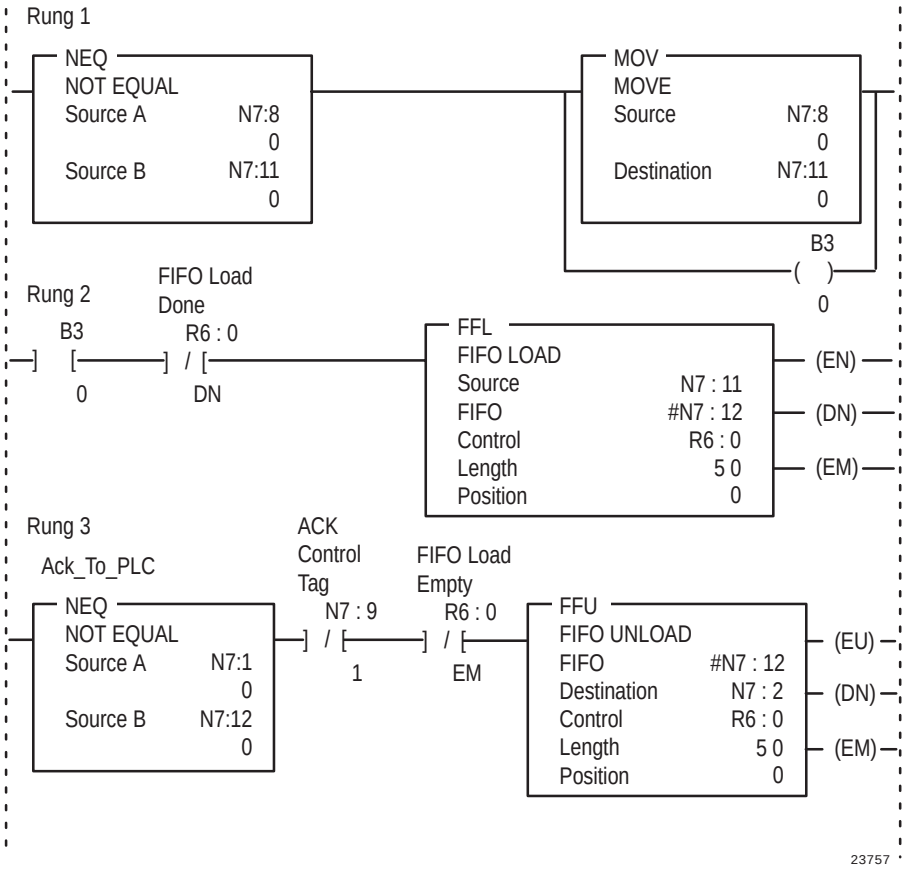
The Remote Acknowledge control is optional. However, in order to remotely acknowledge an alarm, a digital or analog tag or expression must be assigned to this control. Whenever the value of this tag or expression matches an alarm trigger value, the alarm with that trigger value is acknowledged.

All unacknowledged instances of the same alarm are acknowledged at the same time, with the same time stamp. If the Alarm Acknowledge to PLC function is enabled (the Acknowledge to PLC tag is assigned, and the Alarm Acknowledge to PLC option is set to Operator and Remote), it will be activated when a remote acknowledge occurs, even if the operator or the PLC has already acknowledged the alarm. Refer to the sections, “Alarm Acknowledge to PLC” and “Acknowledge to PLC and Acknowledge Control” earlier in this chapter, for more details.

Remote Acknowledge allows only one alarm to be acknowledged at a time. Since alarm acknowledges are not queued, it is recommended that, if you are using remote acknowledge, you also enable the Alarm Acknowledge to PLC function, set the Alarm Acknowledge to PLC to Operator and Remote, and use the Acknowledge Control bit to control the flow of remote alarm acknowledges, as follows:


Waiting for Remote Ack Expression to change
Receive Remote Ack Value from PLC
Is a previous Acknowledge still active?
Yes
No
Does Remote Ack Value match an ACTIVE alarm?
Yes
No
Ack Time Stamp same for ALL ACTIVE instances of alarm in Alarm History file
Is Alarm Window active?
Yes
No
Update Alarms in Alarm Window
Is Alarm History Screen active?
Yes
No
Update Alarms in Alarm History Screen
Does Ack Alarm have Audio Flag?
Yes
No
Disable Beeper if PLC–controlled Audio is not active
Does the Ack Alarm have Relay Flag?
Yes
No
Disable Relay if PLC–contolled Relay is not Active
Does the Ack Alarm have Print Flag?
Yes
No
Print ALL instances of Acknowldged Alarm from the Oldest to Newest
Is Notify PLC option enabled?
Yes
No
Remote Alarm Ack Value=Alarm Normal Value AND Alarm Ack Tag is Block
Yes
No
Write Remote Alarm Ack Value to Alarm Ack Tag (Word or Block)
Was the Write to Alarm Ack out of range?
Yes
No
Generate MISTAKE Ack Out or Range
Set Acknowledge Button to PLC Bit for Hold Time Duration

Example 4: PLC Logic Used to Control Remote Alarm Acknowledge from the PLC to the PanelView Terminal



Comments:

Rung 1: N7:8 contains the trigger value of the alarm to be remotely acknowledged.

Rung 2: The FIFO is loaded as alarm trigger values to be remotely acknowledged are copied from N7:8 to N7:11.

Rung 3: The FIFO is unloaded, resulting in alarms being remotely acknowledged.

N7:1 is the Acknowledge to PLC address. It contains the trigger value of the alarm that was last acknowledged, and is sent from the PanelView terminal to the PLC.

N7:9/1 disables this rung for the time duration set in the Acknowledge Control Hold Time parameter. This allows control of the flow of remote alarm acknowledges.

N7:12 contains the trigger value of the alarm to be remotely acknowledged from the PLC and is unloaded to the Remote Acknowledge tag address, N7:2.

Note: If you use Remote Acknowledge, your application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Silence Alarms to PLC

When the operator presses the Silence button in the Alarm Message Window or the Alarm History screen, the terminal sets the assigned tag to 1 for the Remote Alarm Operation Hold Time and then resets it to 0. This control uses tags only. Use a digital or analog tag.

Note: Hold times are configured in the Timing Parameters tab of the Configure Terminal Setup dialog box. See "Configuring Timing Parameters" in Chapter 12, *Configuring Terminal Setup Options*, for more information about these parameters.

PLC Controlled Silence Alarms

The **PLC Controlled Silence Alarms** option silences the beeper and deactivates the alarm relay when the value associated with this tag or expression changes from 0 to non-zero. You can use a digital or analog tag, or an expression, for this control.

This tag affects the beeper and relay if they are active at the time of 0 to non-zero transition and has no effect on alarms triggered after the transition is made.

PLC Controlled Audio

The **PLC Controlled Audio** option allows the programmable controller to sound the terminal's built-in beeper. You can use a digital or analog tag, or an expression that reads one or more digital or analog tags, for this control. Whenever the value associated with this tag or expression changes from 0 to non-zero, the beeper sounds. When the value is reset to 0, the beeper stops.

The programmable controller can sound the beeper by setting the associated control value, or you can allow the operator to control the beeper from the terminal by assigning this control to a push button on a screen.

Important: The PLC Controlled Audio does not affect any alarm messages you may have configured to sound the beeper. These alarms will sound the beeper, regardless of whether you've enabled or disabled PLC Controlled Audio. However, if the audio is controlled in both alarm messages and by the PLC Controlled Audio, the beeper will not be reset unless both the trigger points are reset.

PLC Controlled Relay

The **PLC Controlled Relay** option allows the programmable controller to trigger the alarm relay. Whenever the value associated with this tag or expression changes from 0 to non-zero, the alarm relay turns on. When the value is reset to zero, the alarm relay turns off. You can use a digital or analog tag, or an expression, for this control.

The programmable controller can turn on the relay by setting the associated control value, or you can allow the operator to control the relay from the terminal by assigning this control to a push button on a screen.

ATTENTION: Do not use the alarm relay for control purposes. All critical controls must be hard-wired for safety.

Note: The PLC Controlled Relay doesn't affect any alarm messages you may have configured to trigger the alarm relay. These alarms will energize the relay regardless of whether you've enabled or disabled the PLC Controlled Relay, or whether this control value is 0 or non-zero. However, if the relay is controlled in both alarm messages and by the PLC Controlled Relay, the relay will not be reset unless both the trigger points are reset.

Message to PLC

When an alarm is triggered, if the **Message to PLC** option is enabled for the alarm message, and the control is assigned, the PanelView terminal sends the message text to the PLC, at the location of this control.

The maximum length of an alarm message is 80 characters. The maximum length of a string tag for DH+, ControlNet, or Remote I/O applications is 82 characters. The message will be sent to the PLC in the following format.

Bit	15 8	7 0
1st word	1st character	2nd character
2nd word	3rd character	4th character
	…	…

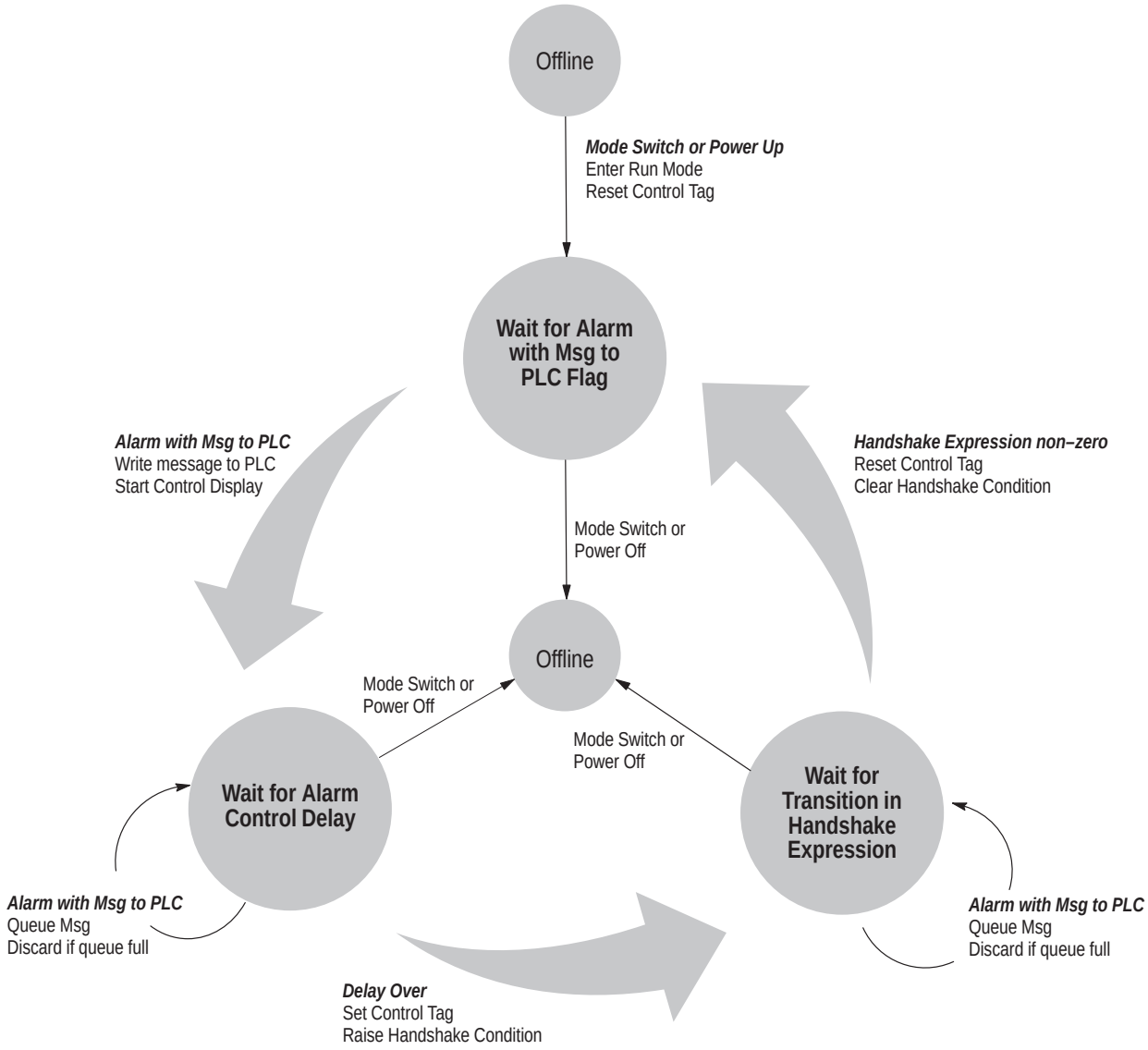
The flow of messages through this tag is controlled using the Message Control and the Message Handshake controls.

The Message to PLC control uses string tags only.

Full Handshaking

When the message text is sent to the PLC, after the Alarm Control Delay interval, the terminal sets the Message Control to 1 and waits for a handshake response from the PLC. When the Message Handshake makes a 0 to non-zero transition, the Message Control tag is reset to 0. If you want to use message handshaking, you must assign both the Message Control and Message Handshake controls, and you must program the PLC to reset the Message Handshake to 0.

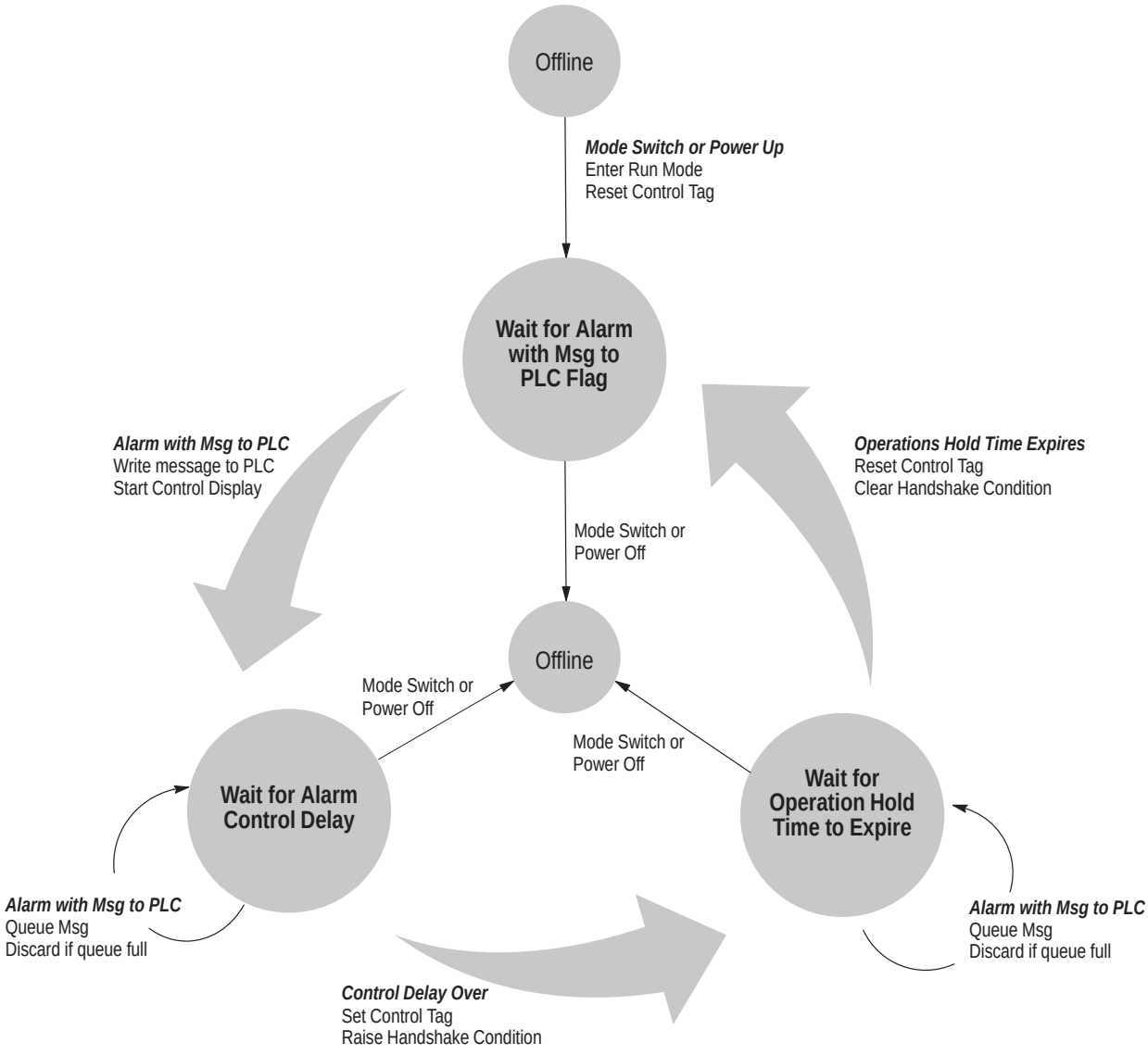
State Transition Diagram for Alarm Message to PLC: Complete Handshaking



Partial Handshaking

If the Message Control is assigned and the Message Handshake is not, the Message Control is set to 1 after the Alarm Control Delay interval, and reset to 0 after the Remote Alarm Operation Hold Time expires.

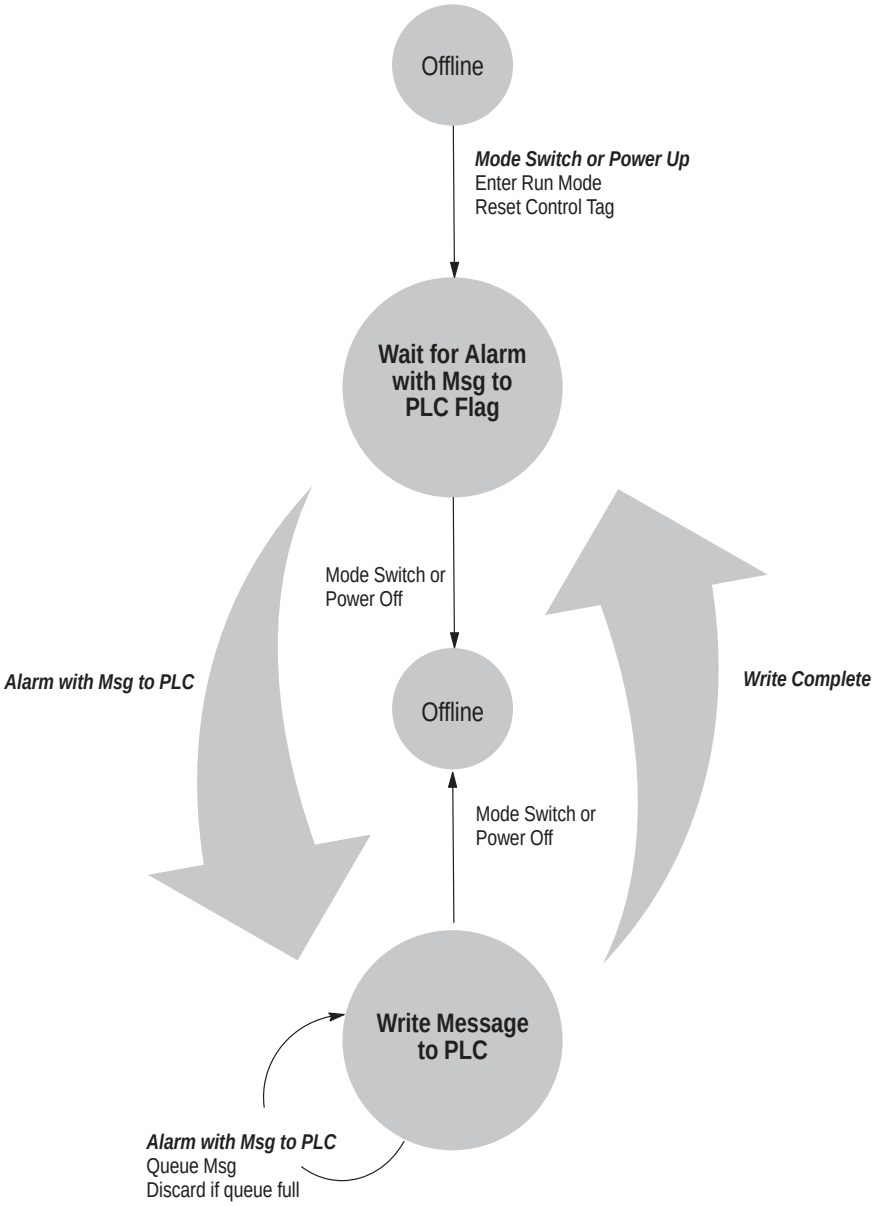
State Transition Diagram for Alarm Message to PLC: Control and No Handshaking



No Handshaking

If the Message Control is not assigned, the message is sent immediately to the PLC, without handshaking to confirm that the PLC receives the message.

State Transition Diagram for Alarm Message to PLC: No Control or Handshaking



The terminal can queue up to 128 Message to PLC requests, in first-in, first-out order. Once the queue is full, write requests will be discarded and a minor fault will appear.

Note: Delay intervals and hold times are configured in the Timing Parameters tab of the Configure Terminal Setup dialog box. See "Configuring Timing Parameters" in Chapter 12, *Configuring Terminal Setup Options*, for more information about these parameters.

Note: If you use the Message to PLC control, your application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Message Control

The **Message Control** is used to control the flow of messages to the PLC. This control uses tags only. Use a digital or analog tag. When an alarm is triggered, the PanelView terminal will initiate handshaking, as described in the previous section, "Message to PLC". The type of handshaking performed depends on whether the Message to PLC, Message Control, or Message Handshake controls are assigned.

Note: Delay intervals and hold times are configured in the Timing Parameters tab of the Configure Terminal Setup dialog box. See "Configuring Timing Parameters" in Chapter 12, *Configuring Terminal Setup Options*, for more information about these parameters.

Note: If you assign a tag to this control your application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Message Handshake

You can use a digital or analog tag, or an expression that reads one or more digital or analog tags, for the **Message Handshake** control. This option provides the terminal with confirmation that alarms with the Message to PLC option enabled are actually received by the PLC. The Message Handshake works with the Message to PLC and Message Control. See "Message to PLC," earlier in this chapter, for more information.

Note: If you use this control your application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions

About the Alarm Message Window

The Alarm Message Window displays the alarm messages you create in the Alarm Messages editor. The Alarm Message Window is triggered by the programmable controller and appears on the terminal when the application is running. It provides messages to the operator, including:

- alarm conditions
- warnings
- informative messages

You create these messages in the Alarm Messages editor. See Chapter 10, *Creating Messages*, for details.

You use the Message Window tab of the Configure Alarms dialog box to enable or disable the Alarm Message Window, to configure the appearance of the window, and to assign control tags or expressions.

You can choose one of these types of Alarm Message Window:

- **Single-line Alarm Message Window** displays only the most recent alarm message at the top or bottom of the screen.
- **Multi-line Alarm Message Window** can display up to 20 single-height or 10 double-height messages.
- **Alarm Message Panel** displays a single message, up to 6 lines long, at the top or bottom of the screen. It can also display the message trigger value.

About Background Alarms

Display of the Alarm Message Window is optional.

If the Alarm Message Window is disabled, alarms will continue to be triggered in the background. They will still cause audio alarms, trip the alarm relay, and send alarm messages to the printer, PLC, and Alarm History screen. However, the Alarm Message Window will not be displayed.

Using the Alarm Message Window in the Terminal

Where the Alarm Message Window appears on the screen, and how it appears, depends on whether it's a Single-line, Multi-line, or Alarm Message Panel.

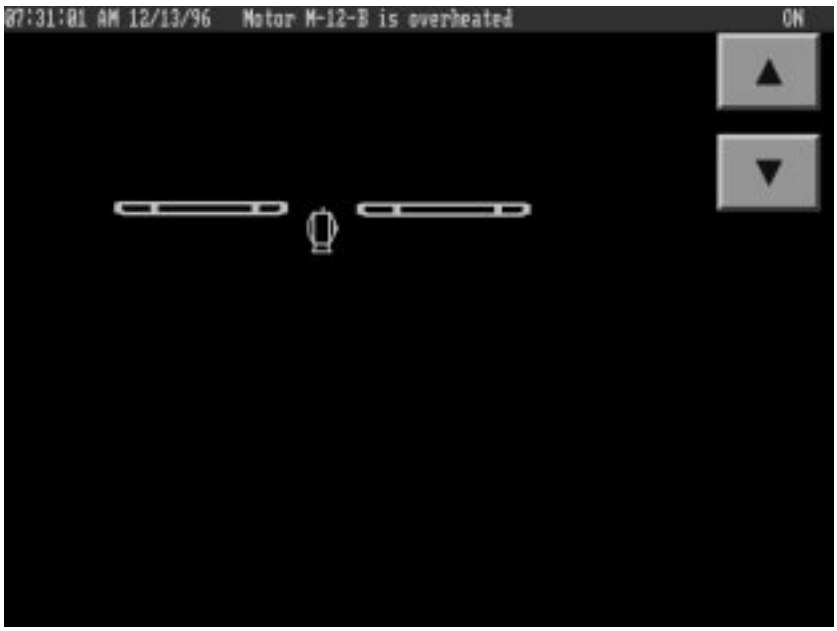
Single-line Alarm Message Window

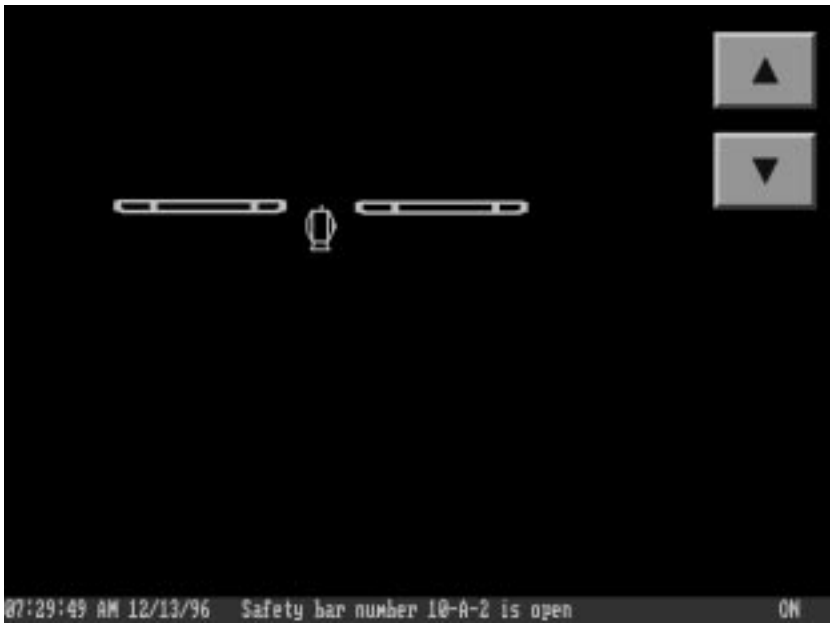
The Single-line Alarm Message Window displays the most recent alarm message only, with the time and date of occurrence and the alarm state. It can appear at the top or bottom of the terminal's screen, depending on how you configure it in the Message Window tab of the Configure Alarms dialog box. This window has no buttons and allows the current application screen's buttons to remain active for operator input.

Tip: Because you won't be able to call up the Alarm History screen from the Single-line Alarm Message Window, include a "Goto Screen" button on your screens that calls up the Alarm History screen.

Important: If the Single-line Alarm Message Window is used, the operator can acknowledge and clear alarms only through the Alarm History screen. See "About the Alarm History Screen," later in this chapter, for more information. Single-line Alarm Message Windows can also be cleared by the programmable controller.

Single-line Alarm Message Window (top of screen)





Single-line Alarm Message Window (bottom of screen)

The Single-line Alarm Message Window can be removed from the screen by:

- acknowledgement of the alarm in the Alarm History screen
- a PLC Controlled Clear Window tag or expression
- a remote acknowledgement from the PLC

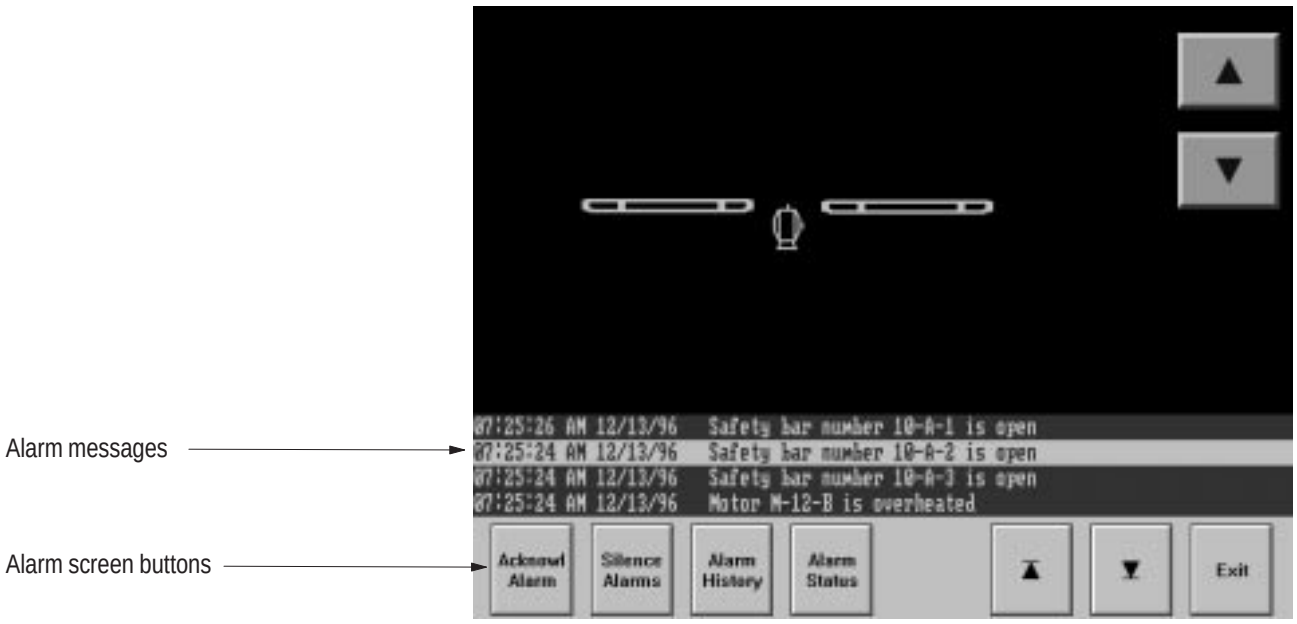
If the alarm displayed in the Single-line Alarm Message Window becomes inactive, it still remains in the window. If the screen changes, the alarm display remains on the new screen, unless it is cleared by the PLC or the alarm is acknowledged by the operator or PLC.

Multi-line Alarm Message Window

The Multi-line Alarm Message Window can display up to 20 single-height or 10 double-height alarm messages, with the time and date of occurrence. The window has a set of buttons with which to respond to alarms. The window always appears at the bottom of the screen.

When the Multi-line Alarm Message Window appears on a keypad terminal, the function keys F1 through F16 are disabled for the application screen until the operator responds to the alarm. On a touch screen terminal, only those touch cells that are covered by the window are disabled for the application screen.

Whether the Multi-line Alarm Message Window covers all or only some of the existing screen depends on how many messages are displayed. The number of messages can be configured.


Alarm messages
Alarm screen buttons
07:25:26 AM 12/13/96 Safety bar number 10-A-1 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-2 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-3 is open
07:25:24 AM 12/13/96 Motor M-12-B is overheated
Acknowl Alarm
Silence Alarms
Alarm History
Alarm Status
Exit

If the programmable controller triggers more alarm messages before the window is cleared, the existing messages shift down, and the new messages are added to the top of the list.

If more messages occur than can fit in the window, the terminal shifts the oldest messages to the Alarm History screen. The “archived messages” icon is displayed between the Alarm Status and Cursor Up buttons to indicate that messages have been moved to the Alarm History screen.

Using Buttons in the Multi-line Alarm Message Window

The buttons in the Multi-line Alarm Message Window control the alarm system as follows:

This button:	Does this:
Acknowl Alarm	Acknowledges, time-stamps, and date-stamps the highlighted alarm message and removes it from the Alarm Message Window. If the alarm message has an audio or relay option enabled, the alarm beeper is silenced or the relay is de-energized. If the alarm message has the print option enabled, the alarm is printed again, with the acknowledgment time and date stamp (the print queue can hold up to 128 triggered and acknowledged alarms). Pressing the **ACKNOWLEDGE ALARM** button has no effect on the audio beeper if the PLC Controlled Audio control value is non-zero, or on the alarm relay if the PLC Controlled Alarm Relay control value is non-zero.

This button:	Does this:
Silence Alarms	Silences the audio indicator and de-energizes the alarm relay if either was triggered by an alarm message. If the Silence Alarms to PLC option is enabled, this button sets the Silence Alarms to PLC tag value to 1 for the configured Remote Alarm Operation Hold Time. Pressing the **SILENCE ALARMS** button has no effect on the audio beeper if the PLC Controlled Audio control value is non-zero, or on the alarm relay if the PLC Controlled Alarm Relay control value is non-zero.
Alarm History	Displays the Alarm History screen.
Alarm Status	Displays the Alarm Status screen.
Cursor Up/Down	Moves the cursor bar up or down the list of alarms, so the operator can acknowledge a specific message.
Exit	Clears the Alarm Message Window. If the Clear Window to PLC option is enabled, pressing **EXIT** sets the Clear Window to PLC tag to 1 for the duration defined as the Remote Alarm Operation Hold Time. If the audio indicator or alarm relay is active, they will be silenced if they are not being activated by PLC Controlled Audio or Alarm Relay functions.

Alarm Message Panel

The Alarm Message Panel displays the most recent alarm message only. Unlike the Single-line Alarm Message Window, the Alarm Message Panel can display a single message, including the Alarm Trigger value, that is up to 6 lines long. Messages are left-aligned in the panel, and may be configured with different font sizes. If the message is longer than a single line, PanelView automatically breaks the message after the last word that will fit on the line, and continues the message on the next line. You can also use the Newline code to specify where you want lines to break, as described in Chapter 10, *Creating Messages*.

The Alarm Message Panel can also display the alarm message trigger value. You can configure the appearance of the message trigger value separately from the appearance of the rest of the Alarm Message Panel. The trigger value can appear before the message and the allowable number of digits is 1 to 11.

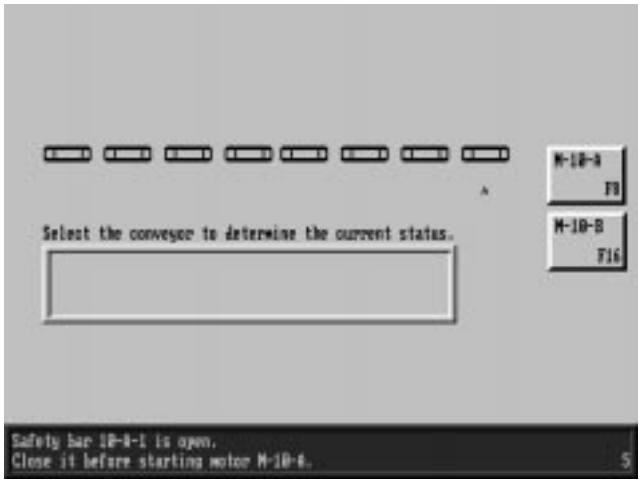
The window can appear at the top or bottom of the terminal's screen, depending on how you configure it in the Message Window tab of the Configure Alarms dialog box. This window has no buttons and allows the current application screen's buttons to remain active for operator input. On PanelView 1200 and 1200e touch screen terminals, if the center of a button is obscured by an Alarm Message Panel, the button will not respond to presses. On PanelView 1400e terminals, if any part of a button is *not* obscured by an Alarm Message Panel, it will still respond to presses.

If you use the Alarm Message Panel the application will be compatible with PanelView Version 2 or later. It will not be compatible with earlier versions.

Tip: Because you won't be able to call up the Alarm History screen from the Alarm Message Panel, include a "Goto Screen" button on your screens to call up the Alarm History screen.

Important: If the Alarm Message Panel is used, the operator can acknowledge and clear alarms only through the Alarm History screen. Alarm Message Panels can also be cleared by the PLC, and the PLC can acknowledge the message that appears in the Alarm Message Panel.

Alarm Message Panel, large text and underlined (top of screen)



Alarm Message Panel, small text (bottom of screen)

The Alarm Message Panel can be removed from the screen by:

- acknowledgement of the alarm in the Alarm History screen
- a PLC Controlled Clear Window tag or expression
- a remote acknowledgment from the PLC

If the alarm displayed in the Alarm Message Panel becomes inactive, the window remains on the screen, but the message is erased. If the screen changes, the alarm display remains on the new screen, unless it is cleared by the PLC or the alarm is acknowledged by the operator or PLC.

Configuring the Alarm Message Window

When you configure the Alarm Message Window you can:

- enable or disable the Alarm Message Window.
- select the window type: single-line, multi-line, or panel.
- determine the location of the window and the number of lines it can display.
- define the appearance of the window—font type, underline, blink, and colors.
- for the panel window type, define whether to display the alarm trigger value, and configure the appearance of the trigger value.
- assign controls to determine how the window is cleared.

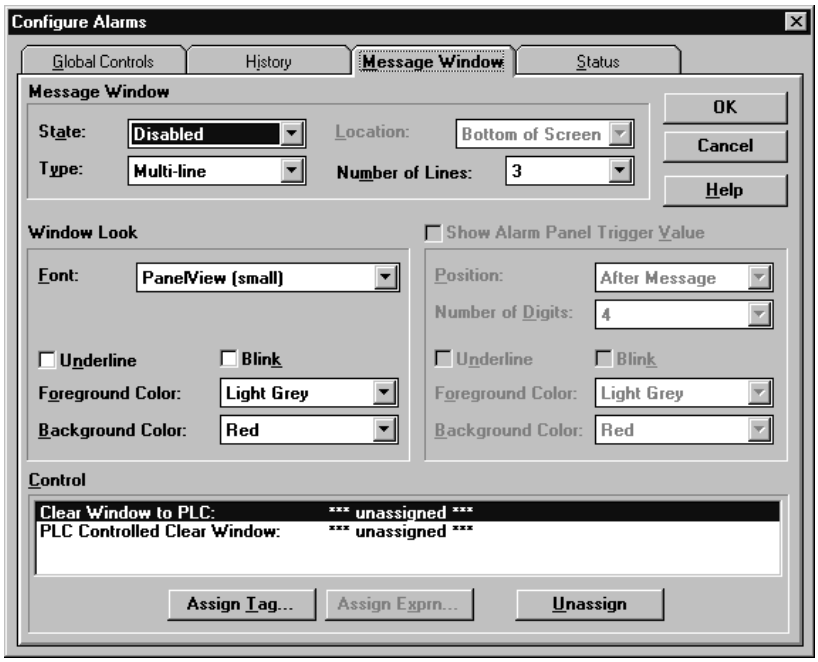
Important: The operator cannot access the Alarm History or Alarm Status screens through the Single-line Alarm Message Window or the Alarm Message Panel. If you choose either of these window types, the operator can call up the Alarm History and Status screens if you design the application with one of these features:

- ensure all screens provide a “Goto Screen” button.
- assign the PLC Controlled Screen Number tag in the PLC I/O Control Options dialog box, then program the PLC to display these screens.

You configure the Alarm Message Window in the Message Window tab of the Configure Alarms dialog box.

To configure the Alarm Message Window:

1. In the Configure Alarms dialog box, choose the Message Window tab.



2. Fill in the following Message Window information:

State
Select Enabled or Disabled in the State drop-down list.

The Alarm Message Window is an optional feature in your application. You can enable the Alarm Message Window if you want it to appear or disable it if you don't. Note that alarms will occur even if the window is disabled. They won't be reported in the Alarm Message Window, but will be displayed in the Alarm History and Alarm Status screens and logged in the Alarm History file.

Type
Select Type and choose Single-line, Multi-line, or Panel.

Location
If you specified a Single-line or Alarm Message Panel, specify its location at the top or bottom of the screen.

Number of Lines
For a Multi-line or Alarm Message Panel, specify the number of lines. The number of lines you choose determines the size of the window on the terminal.

For a Multi-line Alarm Message Window:

- If the window will be using the Small font, you can display from 1 to 20 lines (1 line per message) in the Alarm Message Window.
- If the window will be using the Double High font, you can display from 1 to 10 lines (1 line per message) of alarms in the Alarm Message Window.

For an Alarm Message Panel:

- If the window will be using the Small or Double Wide font, you can display from 1 to 6 lines in the Alarm Message Window (1 message only).
- If the window will be using the Double High or Large font, you can display from 1 to 3 lines in the Alarm Message Window (1 message only).

3. Configure the Window Look as follows:

Font Type

For this window type:	Choose one of these fonts:
Single-line	Small only.
Multi-line	Small or Double High.
Panel	Small, Double Wide, Double High, or Large.

Underline
Click the checkbox to choose Yes, if you want the alarm messages to be underlined. Leave it unchecked to choose No.

Blink
Click the checkbox to choose Yes, if you want the alarm messages to blink. Leave it unchecked to choose No.

Foreground Color
Choose the foreground color of the alarm message.

Background Color
Choose the background color of the alarm message.

4. For an Alarm Message Panel, configure the message trigger value as follows:

Show Alarm Panel Trigger Value
Leave the box checked to display the message trigger value, or clear the checkbox to not display the trigger value. If you choose not to display the trigger value, go to Step 5.

Position
Choose Before Message or After Message to specify where in the panel to display the trigger value. The trigger value is left-justified in the trigger value field if you choose Before Message. It is right-justified if you choose After Message.

Number of Digits
Select how many digits of the trigger value to display: 1 to 11.

Note: If a trigger value requires more digits than the number you select, when that value's message is displayed the trigger value field is filled with asterisks on the PanelView terminal.

Underline
Click the checkbox to choose Yes, if you want the trigger value to be underlined. Leave it unchecked to choose No.

Blink
Click the checkbox to choose Yes, if you want the trigger value to blink. Leave it unchecked to choose No.

Foreground Color
Choose the foreground color of the trigger value.

Background Color
Choose the background color of the trigger value.

5. Assign controls in the Control field.

This control:	Does this:
Clear Window to PLC	The Clear Window to PLC control tells the PLC that the Alarm Message Window is no longer displayed on the PanelView terminal's screen. When the user clears the Single-line, Multi-line or Alarm Message Panel, the terminal sets the Clear Window to PLC control to 1 for the Remote Alarm Ack Control Hold Time. After the Remote Alarm Ack Control Hold Time, the control is reset to 0. The user can clear the Single-line, Multi-line, or Alarm Message Panel by acknowledging all alarms. The user can also press Exit on the Multi-line Alarm Message Window to clear it. Assign a tag to enable this option. **Important:** Only the Multi-line Alarm Message Window contains an Acknowledge or Exit button.
PLC Controlled Clear Window	Clears the Alarm Message Window as well as the alarm beeper and relay when a 0 to non-zero transition is detected on the assigned control. Assign a tag or expression to enable this option. For information about expressions see Chapter 7, *Creating Expressions.* **Note:** Subsequent alarm triggers are processed even if the tag or expression remains at non-zero.

6. Choose OK to save the Message Window configuration.

About the Alarm Status Screen

When the application is running, the operator can call up the Alarm Status screen to view a list of alarms. The Alarm Status screen displays active and inactive alarm states from the programmable controller, the number of times a point has gone into alarm, and the total amount of time a point has been in alarm.

The operator can choose to view all the alarm messages that were created in the Alarm Messages editor, or only the active alarms, or only the past alarms.

Working with the Alarm Status Screen in the Terminal

The following sections explain how the operator can work with the Alarm Status screen when the application is running.

The following figure shows an example of the Alarm Status screen.

The Alarm Status Screen provides the following information about each alarm:

This field:	Contains this information:
Alm Trigger (Alarm Trigger)	The alarm trigger value.
Qty (Alarm Quantity)	How many times the alarm has gone into the alarm state. This field can display a maximum number of 9999 alarms.
Acc Time (Accumulated Time)	The total amount of time each alarm has been on, or in-alarm.
On (Alarm State)	Whether alarms are active (ON), indicated by an asterisk (*), or inactive (OFF), indicated by a blank.
Message	The alarm message text associated with the alarm message number.

The quantity and accumulated time can be reset to 0, either by the operator or programmable controller.

Calling up the Alarm Status Screen from the Terminal

When the application is running, the operator can call up the Alarm Status screen from the Multi-line Alarm Message Window, the Alarm History screen, screen selection objects, or the PLC Controlled Screen Change function. How the terminal display appears depends on how the Alarm Status screen is invoked:

- When called up from the Multi-line Alarm Message Window, it overlays the Alarm Message Window and the currently displayed application screen. The Screen Number to PLC is not changed.
- When called up from the Alarm History screen, the Alarm Status screen replaces it. The Screen Number to PLC is not changed.
- When called up by a screen selection object or via PLC Controlled Screen Change, it will overlay the application screen such that exiting from the Alarm Status screen will reveal the application screen beneath. The Screen Number to PLC, if enabled, will be set to the Alarm Status screen number.

If the Alarm Status screen is invoked by the PLC Controlled Screen Change or a screen selection object, this occurs:

- When the Alarm Status screen is exited, the Screen Number to PLC will be set to that of the application screen previously obscured by the Alarm Status screen.
- Another PLC Controlled Screen change request will remove the Alarm Status screen and the current application screen to display the new screen.

Viewing the Alarm Messages

The Alarm Status screen can hold up to 2000 alarm messages, but only 14 messages can fit on the screen at a time. The first 50 characters of the message appear on screen. The operator can use the Page Up and Page Down buttons to scroll through the messages.

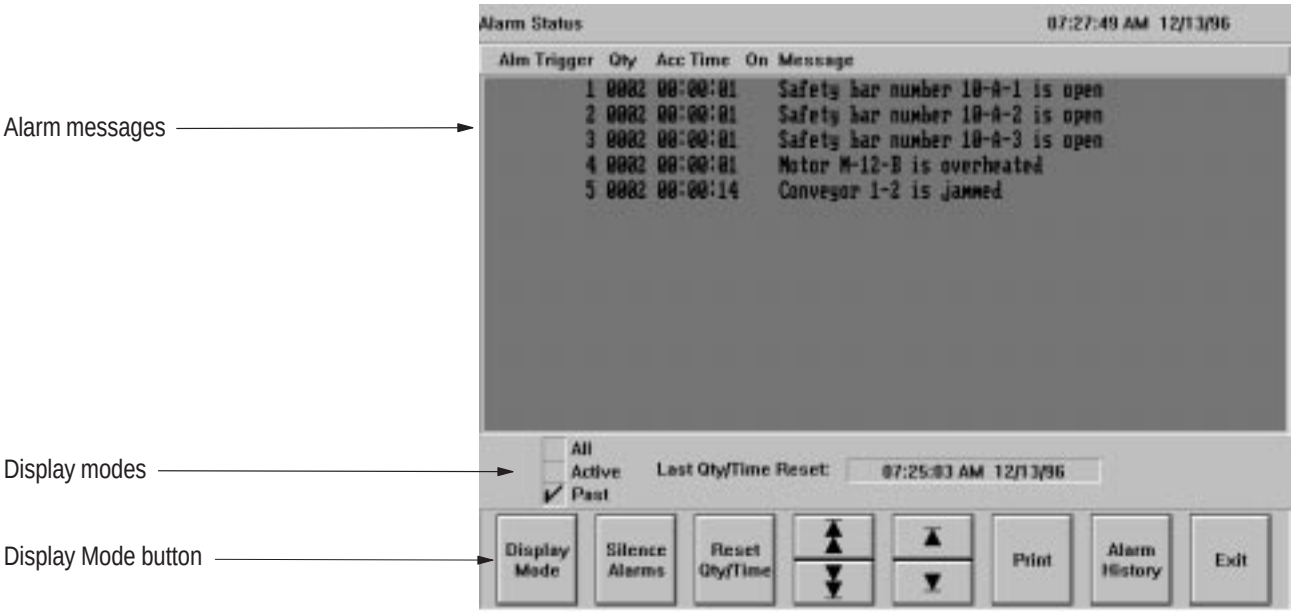
There are three different viewing modes:

- Display All Alarms Mode
- Display Active Alarms Mode
- Display Past Alarms Mode

To change the viewing mode:

▶ Press the Display Mode button in the Alarm Status screen.

Pressing this button changes the display mode. The current display mode is indicated by the checkbox. In the example below, the past alarms are displayed (Display Past Alarms Mode).



Working in Display Active Alarms Mode

The Display Active Alarms Mode shows all the alarms currently active (whose state is ON).

The Alarm Status screen opens first in Active mode. This mode shows all alarms that are ON, in numerical order of the alarm trigger, along with the alarm's quantity, accumulated time, and message information.

If you enter Display Active Alarms mode from Display All Alarms or Display Past Alarms mode, the display starts with the first active alarm from the current page on the screen, and displays the next 14 active alarms. If there are no active alarms from the current page to the end, the search will go backward until it finds 14 alarms, or until it reaches the beginning of the list.

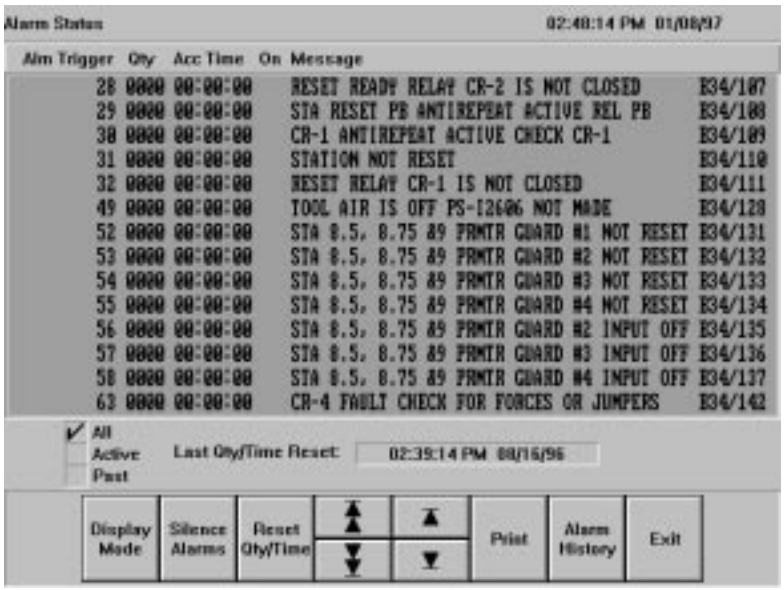
The first alarm in the Active screen acts as an anchor to the alarm messages. If this alarm changes to OFF, all the alarms on the screen shift upwards. If this same alarm becomes active again, it reappears at the top of the list, and the other alarms shift down. As new alarms are entered on the screen, the alarms below the new entry shift downwards.

If all the alarms displayed on a screen turn OFF, and there are no higher alarm numbers ON, the display scrolls to the previous page of alarms.

Working in Display All Alarms Mode

The Display All Alarms Mode lists all alarms, whether they are in-alarm or not. This includes the alarms that haven't been in-alarm since the last reset.

When the operator switches modes from Display Active Alarms or Display Past Alarms to Display All Alarms, the alarm at the top of the screen stays in place and the next 14 alarms are displayed.


Alarm Status
02:40:14 PM 01/08/97
Alm Trigger Qty Acc Time On Message
28 0000 00:00:00 RESET READY RELAY CR-2 IS NOT CLOSED B34/107
29 0000 00:00:00 STA RESET PB ANTIREPEAT ACTIVE REL PB B34/108
30 0000 00:00:00 CR-1 ANTIREPEAT ACTIVE CHECK CR-1 B34/109
31 0000 00:00:00 STATION NOT RESET B34/110
32 0000 00:00:00 RESET RELAY CR-1 IS NOT CLOSED B34/111
49 0000 00:00:00 TOOL AIR IS OFF PS-12606 NOT MADE B34/128
52 0000 00:00:00 STA 8.5, 8.75 &9 PRMTR GUARD #1 NOT RESET B34/131
53 0000 00:00:00 STA 8.5, 8.75 &9 PRMTR GUARD #2 NOT RESET B34/132
54 0000 00:00:00 STA 8.5, 8.75 &9 PRMTR GUARD #3 NOT RESET B34/133
55 0000 00:00:00 STA 8.5, 8.75 &9 PRMTR GUARD #4 NOT RESET B34/134
56 0000 00:00:00 STA 8.5, 8.75 &9 PRMTR GUARD #2 INPUT OFF B34/135
57 0000 00:00:00 STA 8.5, 8.75 &9 PRMTR GUARD #3 INPUT OFF B34/136
58 0000 00:00:00 STA 8.5, 8.75 &9 PRMTR GUARD #4 INPUT OFF B34/137
63 0000 00:00:00 CR-4 FAULT CHECK FOR FORCES OR JUMPERS B34/142
All
Active
Past
Last Qty/Time Reset: 02:39:14 PM 08/16/96
Display Mode
Silence Alarms
Reset Qty/Time
Print
Alarm History
Exit

Working in Past Alarms Mode

The Past Alarms Mode shows all alarms that are active or have been active since the last reset. The Alarm State column shows ON or OFF to indicate the current active alarms.

The first alarm displayed is the first in the current or subsequent pages with a quantity or accumulated time of greater than 0.

The first alarm in the Past Alarms screen acts as an anchor to the screen of alarm messages. When a new alarm (an alarm that has not been active since the last reset) is entered on the screen, the alarms below the new entry shift downwards. If a past alarm with a trigger value below the anchor alarm trigger value becomes active, it will not be displayed.

About Alarm Quantity and Accumulated Time

The Qty and Acc Time fields are calculated differently for bit-triggered and word-triggered alarms.

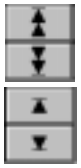
For bit-triggered alarms, the Quantity is the number of times the bit was switched from 0 to 1, and the Acc Time is the amount of time the bit was set as 1.

For word-triggered alarms, the Quantity is the number of times the tag was set to the alarm's trigger value, and the Acc Time is the amount of time the tag was set to the alarm's trigger value. Because you cannot monitor multiple alarms with word-triggered alarms, unless you have one tag for each alarm, the Acc Time is not necessarily the amount of time a particular alarm has been active; it is the amount of time the tag was set to the alarm's trigger value. If multiple alarms are writing to the same tag, multiple alarms will increase their Quantity value, but only the most recent alarm will continue to increase its Acc Time value.

Using the Alarm Status Screen Buttons

The Alarm Status screen has several buttons the operator can use to get information.

This button:	Does this:
Display Mode	Allows the operator to choose one of three viewing modes: all alarms (All), active alarms only (Active), or previous alarms (Past).
Silence Alarms	Silences the beeper and unlatches the alarm relay if either was triggered by an alarm message. It performs the same function as the Silence button in the Alarm Message Window.
Reset Qty/Time (if enabled)	Allows the operator to reset the alarm count to 0, and the accumulated time in-alarm to 00:00:00. User confirmation is required. For bit-triggered alarms, after a reset, an active alarm that has not been re-triggered will continue to log accumulated time; the alarm will be displayed with Qty=0 and a non-zero accumulated time. For word-triggered alarms, the accumulated time data is not displayed, so the display shows Qty=0 only.
Page Up/Page Down	Displays the previous or following page of alarms.
Cursor Up/Cursor Down	Moves up or down one line through the list of alarms.
Print	Prints out the alarm status of all the alarms in the currently active screen mode. User confirmation is required.
Alarm History	Opens the Alarm History screen.
Exit	Depending on how the Alarm Status screen was initially invoked, this button returns the operator to the Alarm History screen, the Alarm Message Window, or the last user-defined application screen.



Important: Whether the alarm quantity and accumulated time counts are reset manually by the operator from the Alarm Status screen or automatically by the programmable controller depends on how the screen is configured.

Resetting the Alarm Status Screen's QTY/TIME

The Alarm Status screen's quantity and time counts (QTY/TIME) are reset when the PLC Controlled QTY/TIME Reset tag or expression changes from 0 to non-zero. Alarm monitoring continues even though the tag value is non-zero. The operator can also reset the Alarm Status screen's QTY/TIME, depending on how the screen is configured.

Important: When the Alarm Status screen is reset, all the Alarm Quantity and Accumulated Time fields in the Alarm Status screen are reset to 0.

The programmable controller can reset the Alarm Status screen only if the PLC Controlled QTY/TIME Reset control is assigned. The operator can reset the Alarm Status screen by:

- pressing, and confirming the Reset Qty/Time button (if enabled)
- downloading or selecting a new application in the terminal

The Last Qty/Time Reset stamp is retentive. When an application is first run, it will be set to the time the terminal is switched to Run Mode.

What Happens when the Operator Resets the QTY/TIME?

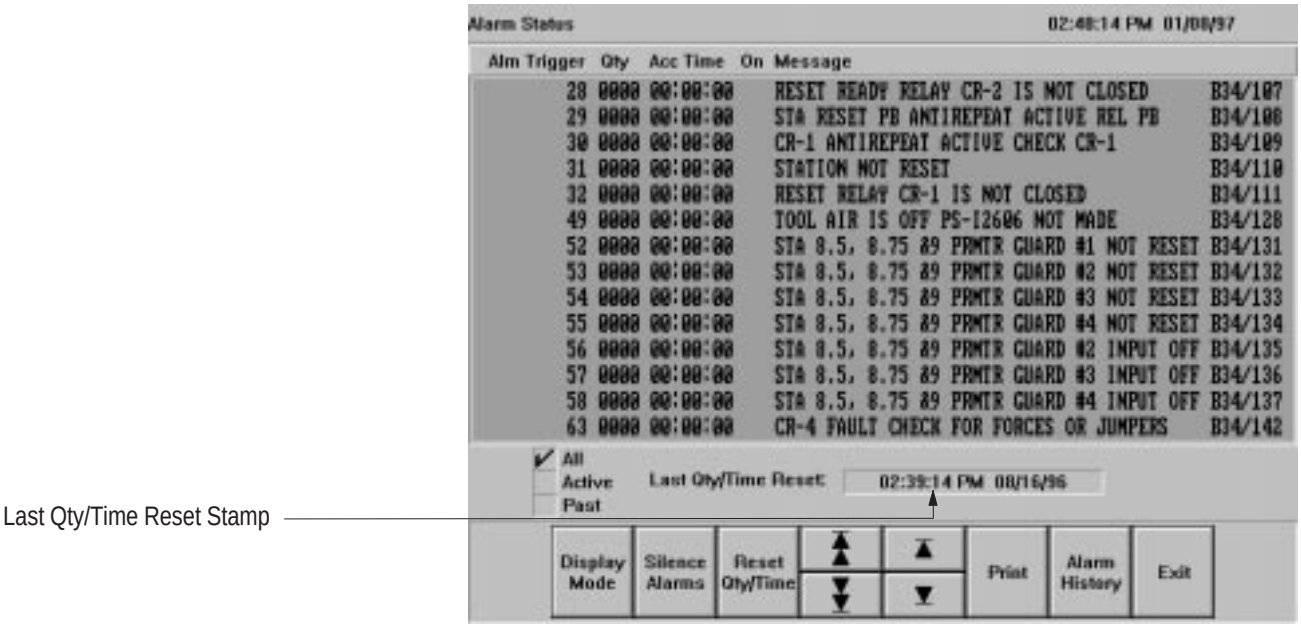
When a reset occurs, the terminal automatically sets the Alarm Status screen's QTY/TIME Reset to PLC tag (if assigned). It sets this tag for the Remote Alarm Operation Hold Time after detecting the Alarm Status reset, whether or not the Reset QTY/TIME button is still being pressed or the PLC Controlled QTY/TIME Reset tag or expression is still active.

The Reset QTY/TIME button (or the PLC Controlled QTY/TIME Reset tag or expression) must make another 0 to non-zero transition before the Alarm Status screen's QTY/TIME Reset to PLC tag is set.

Important: While the Alarm Status screen's QTY/TIME Reset to PLC tag is set to 1, any subsequent Alarm Status QTY/TIME resets (from the PLC or via the Reset QTY/TIME button) will not affect the state of this tag.

Important: Make sure the Reset QTY/TIME button is enabled or the PLC Controlled QTY/TIME Reset control is assigned. If both are disabled, the alarm count cannot be reset.

When a reset occurs, the Alarm Status screen updates the Alarm Status Last Qty/Time Reset stamp.


Alarm Status
02:48:14 PM 01/08/97
Alm Trigger Qty Acc Time On Message
28 0000 00:00:00 RESET READY RELAY CR-2 IS NOT CLOSED B34/107
29 0000 00:00:00 STA RESET PB ANTIREPEAT ACTIVE REL PB B34/108
30 0000 00:00:00 CR-1 ANTIREPEAT ACTIVE CHECK CR-1 B34/109
31 0000 00:00:00 STATION NOT RESET B34/110
32 0000 00:00:00 RESET RELAY CR-1 IS NOT CLOSED B34/111
49 0000 00:00:00 TOOL AIR IS OFF PS-12606 NOT MADE B34/128
52 0000 00:00:00 STA 8.5, 8.75 89 PRMTR GUARD #1 NOT RESET B34/131
53 0000 00:00:00 STA 8.5, 8.75 89 PRMTR GUARD #2 NOT RESET B34/132
54 0000 00:00:00 STA 8.5, 8.75 89 PRMTR GUARD #3 NOT RESET B34/133
55 0000 00:00:00 STA 8.5, 8.75 89 PRMTR GUARD #4 NOT RESET B34/134
56 0000 00:00:00 STA 8.5, 8.75 89 PRMTR GUARD #2 INPUT OFF B34/135
57 0000 00:00:00 STA 8.5, 8.75 89 PRMTR GUARD #3 INPUT OFF B34/136
58 0000 00:00:00 STA 8.5, 8.75 89 PRMTR GUARD #4 INPUT OFF B34/137
63 0000 00:00:00 CR-4 FAULT CHECK FOR FORCES OR JUMPERS B34/142
All
Active
Past
Last Qty/Time Reset:
02:39:14 PM 08/16/96
Display Mode
Silence Alarms
Reset Qty/Time
Print
Alarm History
Exit

Last Qty/Time Reset Stamp

The Last Qty/Time Reset stamp is always retained between power cycles.

What Happens when the Terminal is Off Line?

The terminal stops incrementing the accumulated alarm time and quantity when it is offline to the programmable controller. In this case, “offline” means one or more of the following conditions exists:

- the terminal is disconnected from the controller
- the terminal is in Configuration mode
- the controller is not in Run mode
- the controller has the terminal Remote I/O racks disabled

When the terminal goes back online, the alarm counts are increased by one for those alarms currently in alarm, and accumulation of the times resumes.

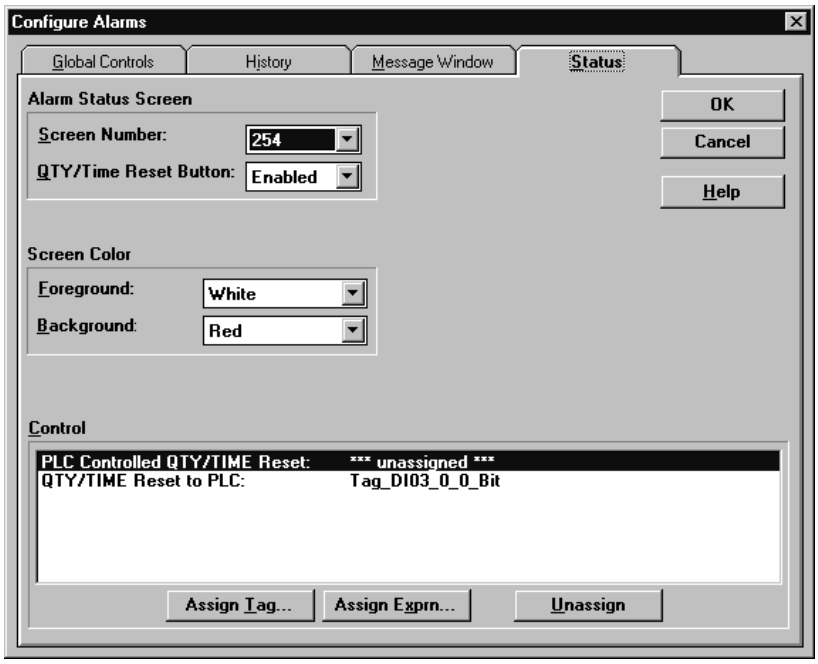
Configuring the Alarm Status Screen

You can configure the Alarm Status screen as follows:

- Assign the screen number.
- Enable or disable the QTY/Time Reset Button.
- Set the colors.
- Assign the PLC Controlled QTY/TIME Reset control to enable the option. When enabled, this option allows the PLC to reset the alarm count and total time in-alarm to zero.
- Assign the QTY/TIME Reset to PLC control. This tag indicates to the PLC when an Alarm Status QTY/TIME Reset has occurred. This can be done either with the QTY/Time Reset Button, or through the PLC.

To configure the Alarm Status screen:

1. In the Configure Alarms dialog box, choose the Status tab.



2. Fill in the following Alarm Status screen information:

   Screen Number
   Select the screen number from the drop-down list. You can assign any unused screen number from 1 to 255.

   Important: If you leave the Screen Number field blank, the Alarm Status screen cannot be called up by screen select objects or by a PLC Controlled screen change.

QTY/Time Reset Button
Choose Enabled or Disabled from the drop-down list. When you enable the QTY/Time Reset Button option, the QTY/Time Reset Button will appear on the Alarm Status screen. By pressing this button, the operator can reset the alarm count and total time in alarm to zero.

There are two ways to have the quantity and time values reset on the Alarm Status screen:

- Manual reset—the operator can reset the alarms by pressing the QTY/Time Reset Button if the QTY/Time Reset Button is enabled.
- Automatic reset—the programmable controller can reset the alarms automatically if the PLC Controlled QTY/TIME Reset control is assigned.

  Important: Be sure you have at least one reset enabled. If both are disabled, the alarm count can't be reset.

Note: If the value of the TIME field exceeds 99:99:99, the field displays **:**:** instead. The QTY field stops accumulating at 9999.

Foreground
Choose the foreground color for the messages in the Alarm Status screen.

Background
Choose the background color for the messages in the Alarm Status screen.

3. Assign controls in the Control field.

This control:	Does this:
PLC Controlled QTY/TIME Reset	Allows the programmable controller to reset the time and quantity counts in the Alarm Status screen. Whenever the programmable controller sets the control value from 0 to non-zero, the time and quantity are reset. Assign a tag or expression to enable this option. For information about expressions see Chapter 7, *Creating Expressions*.
QTY/TIME Reset to PLC	Allows the terminal to notify the programmable controller when an Alarm Status QTY/Time Reset has occurred, whether from a PLC Controlled QTY/TIME Reset, or from the Reset QTY/Time button. When a reset occurs, the control is set to 1 for the Remote Alarm Operation Hold time before it is reset to 0. Assign a tag to enable this option.

See Chapter 6, *Defining Tags*, and Chapter 7, *Creating Expressions*, for information about how to assign and unassign tags and expressions.

4. Choose OK to save the Alarm Status screen configuration.

About the Alarm History Screen

The operator can view a list of historical alarms by calling up the Alarm History screen. The Alarm History screen contains a list of the most recent alarms (maximum 128) for the current application, with the most recent alarm at the top of the list.

Note: If more than 128 alarms occur, the oldest alarm records will be dropped from the list. The maximum number of alarm triggers and acknowledgements that can be in the print queue at one time is also 128. If you need to keep a permanent list of old alarms, print the message list regularly, or upload the Alarm History file. See the *PanelBuilder 1400e Transfer Utility User Manual* for information about uploading the Alarm History file.

Working with the Alarm History Screen in the Terminal

This section tells you how the Alarm History screen works when the application is running.

Calling up the Alarm History Screen from the Terminal

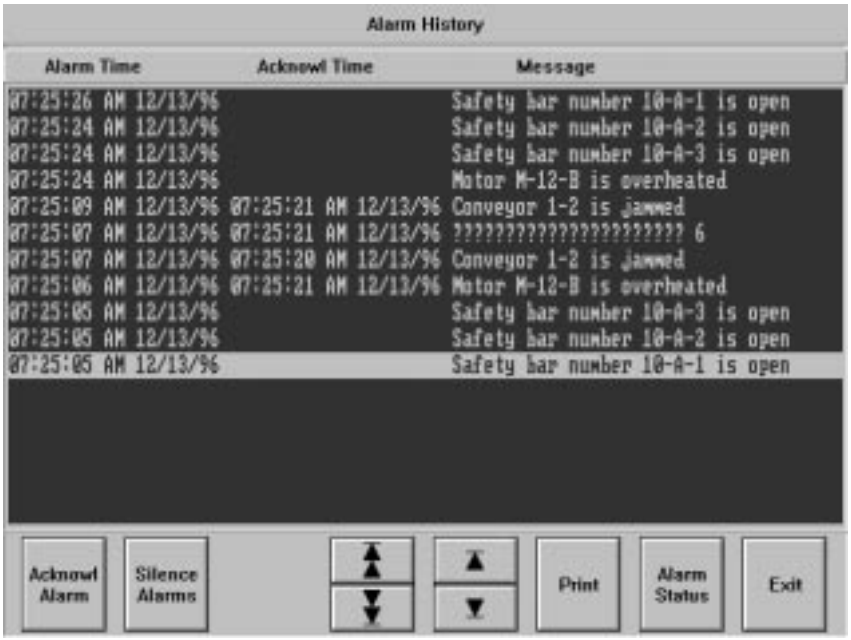
When the application is running, the operator can call up the Alarm History screen from the Multi-line Alarm Message Window, the Alarm Status screen, screen selection objects, or the PLC Controlled Screen Change function. How the terminal display appears depends on how the Alarm History screen is invoked:

- When called up from the Multi-line Alarm Message Window, it overlays the Alarm Message Window and the currently displayed application screen. The Screen Number to PLC is not changed.
- When called up from the Alarm Status screen, it replaces it. The Screen Number to PLC is not changed.
- When called up by a screen selection object or via PLC Controlled Screen Change, it overlays the application screen such that exiting from the Alarm History screen will reveal the application screen beneath. The Screen Number to PLC, if enabled, is set to the Alarm History screen number.

If the Alarm History screen is invoked by the PLC Controlled Screen Change or a screen selection object, this occurs:

- When the Alarm History screen is exited, the Screen Number to PLC is set to that of the application screen previously obscured by the Alarm History screen.
- Another PLC Controlled Screen change request will remove the Alarm History screen and the current application screen to display the new screen.

The following figure shows an example of the Alarm History screen.


Alarm History
Alarm Time
Acknowl Time
Message
07:25:26 AM 12/13/96 Safety bar number 10-A-1 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-2 is open
07:25:24 AM 12/13/96 Safety bar number 10-A-3 is open
07:25:24 AM 12/13/96 Motor M-12-B is overheated
07:25:09 AM 12/13/96 07:25:21 AM 12/13/96 Conveyor 1-2 is jammed
07:25:07 AM 12/13/96 07:25:21 AM 12/13/96 ?????????????????????? 6
07:25:07 AM 12/13/96 07:25:20 AM 12/13/96 Conveyor 1-2 is jammed
07:25:06 AM 12/13/96 07:25:21 AM 12/13/96 Motor M-12-B is overheated
07:25:05 AM 12/13/96 Safety bar number 10-A-3 is open
07:25:05 AM 12/13/96 Safety bar number 10-A-2 is open
07:25:05 AM 12/13/96 Safety bar number 10-A-1 is open
Acknowl Alarm
Silence Alarms
Print
Alarm Status
Exit

Using Buttons on the Alarm History Screen

This button:	Does this:
Acknowl Alarm	Acknowledges the highlighted alarm. The operator can acknowledge alarms that haven't been acknowledged in the Alarm Message Window. If the alarm message has an audio or relay option enabled, the alarm beeper is silenced or the relay is deactivated. If the alarm message has the print option enabled, the alarm is printed again, with the acknowledgment time and date stamp. Pressing the **Acknowl Alarm** button has no effect on the audio beeper if the PLC Controlled Audio value is non-zero, or on the alarm relay if the PLC Controlled Alarm Relay value is non-zero.
Silence Alarms	Silences the audio indicator and unlatches the alarm relay if either was triggered by an alarm message. It performs the same function as the Silence button in the Alarm Message Window.
Page Up/Page Down	Displays the previous or following page of alarms.
Cursor Up/Cursor Down	Moves the highlight bar up or down one line through the list of unacknowledged alarms so the operator can acknowledge the appropriate alarm. When the highlight bar gets to the bottom or top of the screen, the next press will scroll the highlight bar up or down to the next alarm (if there is one).
Print	Sends the Alarm History information to a printer via the terminal's RS-232 port.
Alarm Status	Opens the Alarm Status screen.
Exit	Returns you to the screen from which you entered the Alarm History screen.

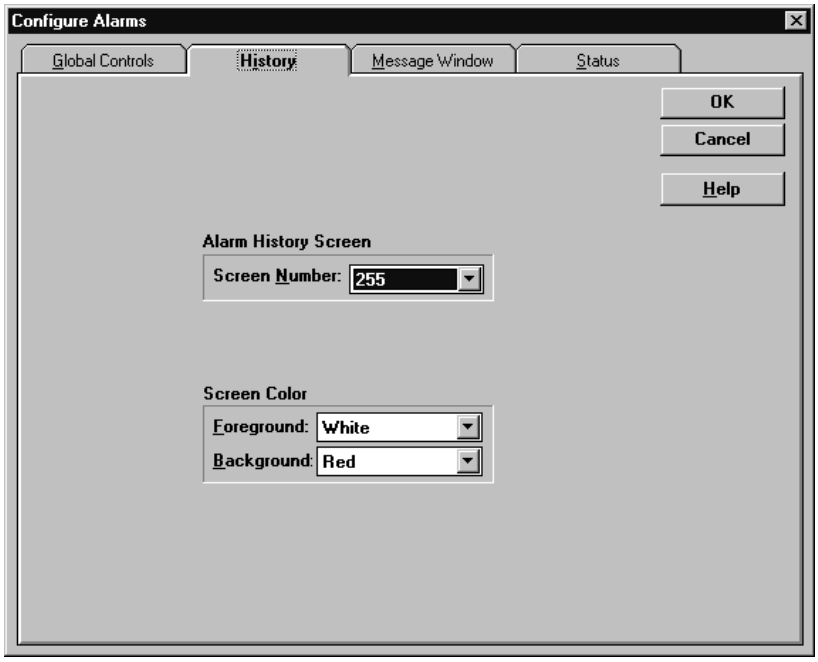
When the operator acknowledges the alarm, the time and date of the acknowledgment are displayed directly to the right of the time and date of the alarm occurrence.

Note: The alarm remains active until the alarm condition no longer exists. Acknowledging the alarm does not deactivate it.

Configuring the Alarm History Screen

To configure the Alarm History screen:

1. In the Configure Alarms dialog box, choose the History tab.



2. Fill in the following information:

   Screen Number
   Select a screen number ranging from 1 to 255 from the drop-down list. The default is 255. You can assign any unused screen number to the Alarm History screen.

   Important: If you leave the Screen Number field blank, the Alarm History screen cannot be called up by screen select objects or by a PLC Controlled Screen Change.

   Foreground
   Choose the foreground color of the Alarm History screen.

   Background
   Choose the background color of the Alarm History screen.

3. Choose OK to save your Alarm History screen configuration.

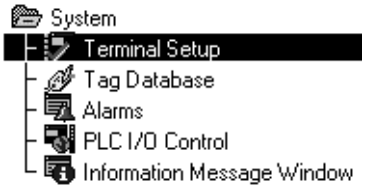
Configuring Terminal Setup Options

This chapter explains how to match the application to your terminal. Instructions include:

- how to specify the Application Startup screen
- how to change the application type
- how to assign a comment to the application file
- how to specify the target PanelView firmware version
- how to select timing parameters, such as Push Button hold time and Remote Alarm hold times and delay
- how to specify default behavior of objects in the application

About Terminal Setup Options

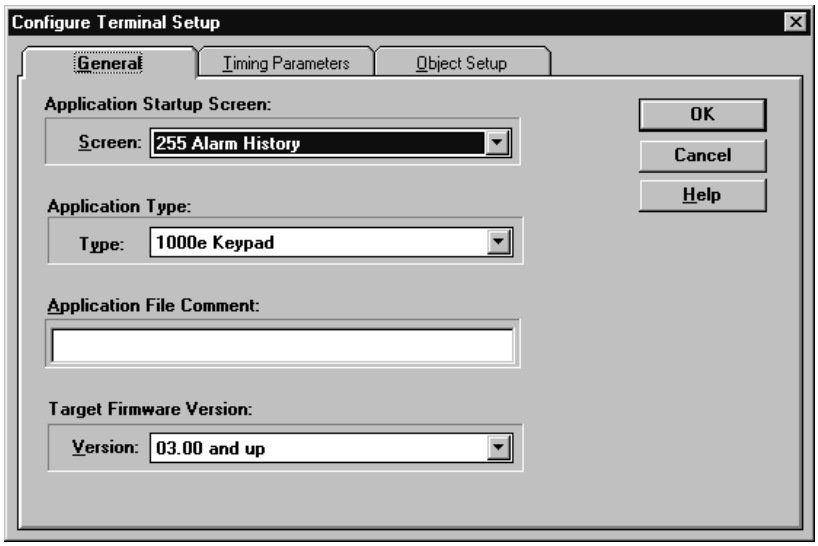
The System folder contains the Configure Terminal Setup dialog box, where you can specify general and timing parameters for the application file and terminal. An example of a parameter would be the push button time required for the PLC to register button presses. You can also specify the default behavior of objects in the application.



To open the Configure Terminal Setup dialog box:

1. Open the application you want to edit.
2. Open the System folder and choose Terminal Setup, or choose from the toolbar.

   The Configure Terminal Setup dialog box appears with two tabs: General and Timing Parameters. For keypad applications, the Configure Terminal Setup dialog box has a third tab—the Object Setup tab.



General Parameters

The General tab is where you specify the Application Startup screen and change the application type. You also use this tab to define a comment to be stored with the file and specify the version of PanelView firmware on which you want the application to run.

The Application Startup Screen

The Application Startup screen is the first screen displayed on the terminal when the application starts up. In a new application, the Application Startup screen is preset to be the Alarm History screen (screen 255, by default). If you want to use a different screen as the Application Startup screen, select a new screen from the Screen drop-down list.

For more information see “Assigning the Application Startup Screen” in Chapter 8, *Creating Screens*.

Changing the Application Type

You select the type of PanelView terminal on which the application is run when you create the application. However, you can use the General tab of the Configure Terminal Setup dialog box to change the application type so it can run on a different terminal type.

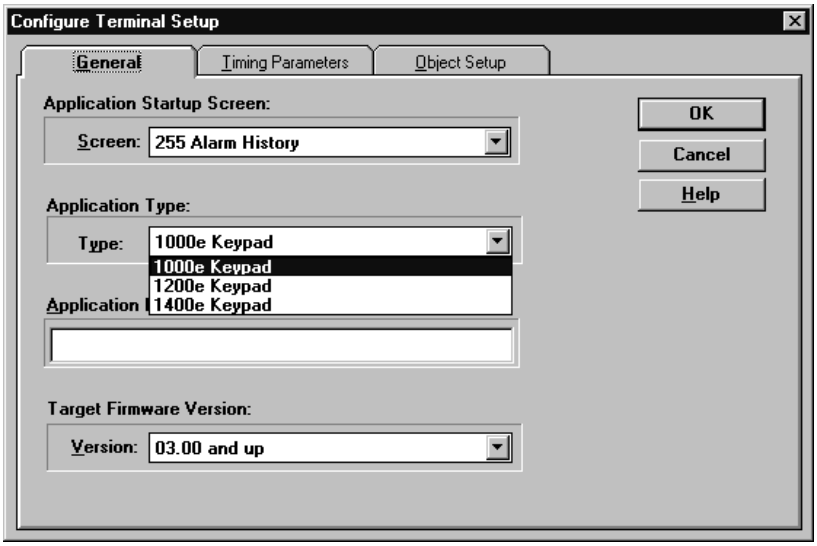
Keep the following in mind when changing the Application Type:

- You can’t run a keypad application on a touch screen terminal, or a touch screen application on a keypad terminal. Once you choose touch screen or keypad for your application type you cannot change to the other type.

- A 1000e or 1400e touch screen application can run interchangeably only on 1000e or 1400e touch terminals. However, you can convert a 1000e or 1400e touch screen application to a 1200e touch screen application (or vice-versa); if you do, check your screens carefully as the size and placement of objects may change.

To change the application type:

▶ Select the application type from the Application Type drop-down list.



Describing the Application

In addition to naming your application by giving it a file name when you first save it, you can enter a brief description of the application.

This description is useful for explaining the function of the application. For example, the description appears on the PanelView Terminal Information and Select Application screens. This helps the operator ensure that the correct application is being selected and executed. The description is also useful for application reports, and doesn't increase the amount of memory used by the application.

To describe the application:

1. Type a description of your application, up to 24 characters in length, in the Application File Comment field.
2. Choose OK.

Target Firmware Version

You can select the version of PanelView firmware that you plan to use when you run your application. The application will be compatible with the Target Firmware Version you specify, as well as with all later versions.

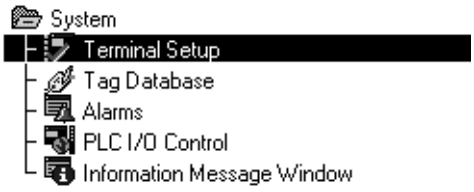
When you validate your application, PanelBuilder warns you if a feature you have used will make the application incompatible with the Target Firmware Version. See Appendix E, *Version Control*, for a list of features that make applications incompatible with earlier versions of the PanelView firmware.

Specifying Target Firmware Version

You can select the version of PanelView firmware that you plan to use when you run your application. The Target Firmware Version you choose allows your application to run on the listed version, as well as later versions of the firmware:

- **Version 1 and up**
- **Version 1.03 and up** allows your application to run on a PanelView terminal connected to the ControlNet communication network
- **Version 2 and up** allows you to use the new features that come with PanelBuilder, Version 2
- **Version 3 and up** allows you to use the new features that come with PanelBuilder, Version 3
- **Version 4 and up** allows you to use the new features that come with PanelBuilder, Version 4

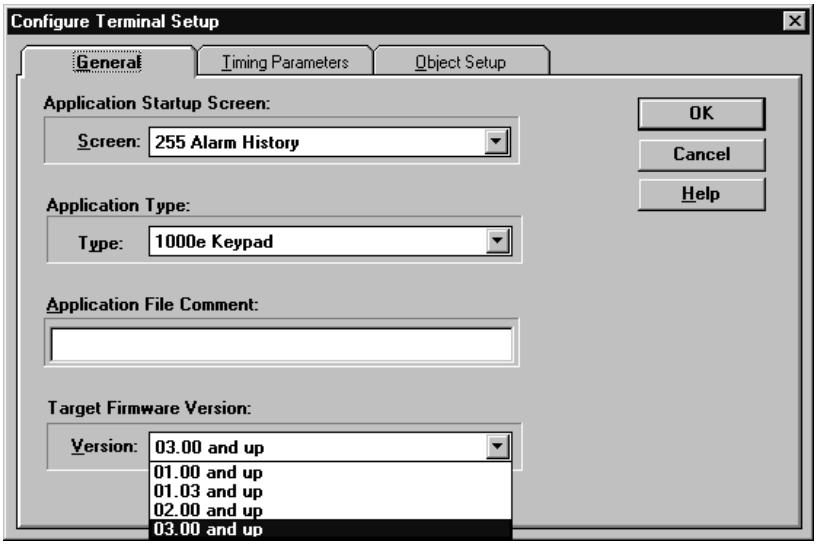
When you validate your application, PanelBuilder warns you if a feature you have used will make the application incompatible with the Target Firmware Version. See Appendix E, *Version Control*, for a list of features that make applications incompatible with earlier versions of the PanelView firmware.



To specify the Target Firmware Version:

1. Open the System folder and choose Terminal Setup, or choose from the toolbar.

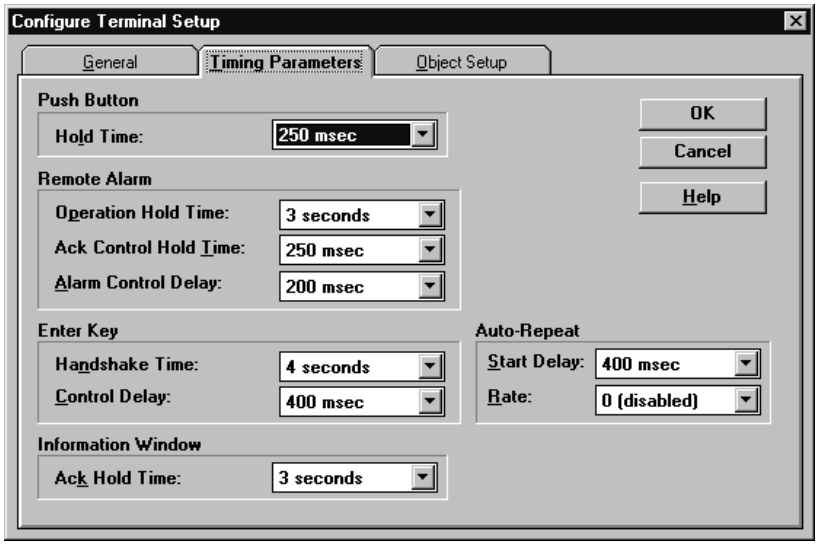
   The Configure Terminal Setup dialog box appears.



2. In the Target Firmware Version field, select from the drop-down list.
3. Choose OK.

Timing Parameters

In the Timing Parameters tab of the Configure Terminal Setup dialog box, you define timing parameters that affect the operation of the terminal and control the flow of information to and from the PLC.



Setting the Push Button Hold Time

If an operator presses a button very quickly, the programmable controller may miss that press between I/O scans. By setting the **Push Button Hold Time**, you set the minimum time that a push button is "held on" by the terminal, so that the programmable controller can sense it.

To set the Push Button Hold Time:

- In the Push Button Hold Time drop-down list, select the minimum time you want to assign.

By default, this is 250 milliseconds. By increasing the "On" time, you ensure that the PLC controller doesn't miss a button press.

For example, if an operator presses and releases a button very quickly, the programmable controller is more likely to register the press if the Push Button Hold Time is set to one second.

Note: If you've set the hold time for one second, operators must wait one second before they can press the same button again. If they press the same button sooner, the terminal will ignore the press.

No operator-initiated screen changes are allowed when a Push Button Hold Time is active. However, PLC-controlled screen changes will cancel the hold time and be processed.

Setting the Remote Alarm Parameters

The **Operation Hold Time** allows you to specify the interval of time the Alarm Control Tag is set for when any of the following is activated:

- Silence Alarms to PLC
- Clear Window to PLC
- QTY/TIME Reset to PLC
- Message to PLC

By default, the interval is 3 seconds.

Note: For Remote I/O applications, this default is usually sufficient. For DH+ and ControlNet applications, the hold time should reflect the highway traffic.

To set the Operation Hold Time:

▶ In the Operation Hold Time drop-down list, select the minimum time you want to assign.

The **Ack Control Hold Time** determines the time interval for the Acknowledge Control. When an alarm is acknowledged by the operator or by the PLC, the address corresponding to this tag is set to 1 for the specified time.

To set the Acknowledge Control Hold Time:

▶ In the Ack Control Hold Time drop-down list, select the minimum time you want to assign.

The **Alarm Control Delay** determines the interval between the time the acknowledged alarm value is sent to the programmable controller and the time the Acknowledge Control tag is set. The delay ensures that the acknowledged alarm value is recorded in the programmable controller before it sees the button tag trigger. The delay controls the flow of alarm messages and acknowledgements to the PLC. If the Alarm Message to PLC is enabled, it also determines the interval between the time a message is sent to the PLC and the time the Message Control tag is set.

To set the Alarm Control Delay:

▶ In the Alarm Control Delay drop-down list, select the time you want to assign.

Note: For more information on alarms, see Chapter 11, *Configuring Alarms*.

Setting the Enter Key Parameters

Many objects, such as the Control List Selector and the Numeric Input Keypad, have an Enter Key Handshake control tag or expression. The following settings apply to those controls.

The **Enter Key Handshake Time** determines how long the terminal will wait for a handshake from the PLC controller.

To set the Enter Key Handshake Time:

- In the Handshake Time drop-down list, select the time you want to assign.

The **Enter Key Control Delay** determines how long the terminal will wait after the Enter key is pressed to send a control value to the programmable controller, and the time the Enter Key Control tag is set to 1. The delay ensures that the new value is recorded in the programmable controller before it sees the button tag trigger.

To set the Enter Key Control Delay:

- In the Control Delay drop-down list, select the time you want to assign.

Setting the Information Message Window Acknowledge Hold Time

When you configure the Information Message Window, you specify how the window is to be removed. One option is Operator Acknowledged. If you choose this option, you can define the Acknowledge to PLC Controller tag.

The **Acknowledge Hold Time** determines how long the address at the Acknowledge to PLC Controller tag will be held on when the operator presses the Acknowledge button.

To set the Acknowledge Hold Time :

- In the Ack Hold Time drop-down list, select the time you want to assign.

Setting the Auto-Repeat Functions

When the operator presses and holds down a button with auto-repeat capability, the button goes into auto-repeat mode. That is, the terminal acts as if the operator were repeatedly pressing and releasing the button, instead of just holding it. You can set both the delay before the auto-repeat mode begins and the number of times per second the press/release action is repeated.

For some objects, you configure the auto-repeat delay and rate when you define the button. These are objects that directly write to a tag without any user confirmation (for example, the Increment/Decrement Value Button and the Control List Selector without Enter Key). Configuring auto-repeat allows more precise control for each object to prevent “overshooting” target values.

When you set the Auto-Repeat values in the Timing Parameters tab of the Configure Terminal Setup dialog box, you set the values for the cursor movement buttons in the following objects:

- Screen List Selector
- Control List Selector with Enter Key
- Cursor Point (in a keypad application) for navigating between cursor points
- Scrolling List’s Cursor List

To change the Auto-Repeat Start Delay:

▶ In the Auto-Repeat Start Delay drop-down list, select the time you want.

The default rate for auto-repeat is 0, which means no repeats. The maximum repeat rate you can assign is 20 repeats per second.

To change the cursor Auto-Repeat Rate:

▶ In the Auto-Repeat Rate drop-down list, select the cursor auto-repeat rate you want.

Object Setup

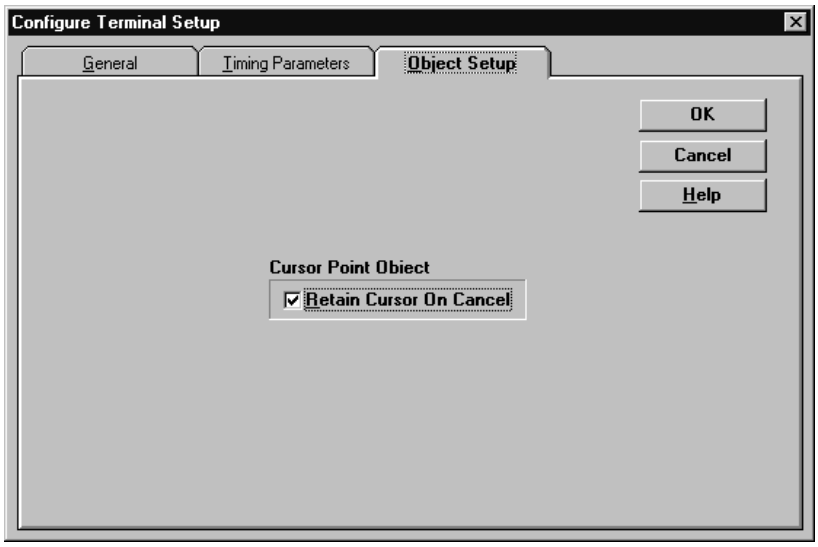
The Object Setup tab allows you to configure the default behavior of objects in the application.

There is currently only one object configuration option—Retain Cursor On Cancel. This option applies to all Numeric Input Cursor Point objects and Set Bit Cursor Point objects in a keypad application (Numeric Input Cursor Point objects and Set Bit Cursor Point objects cannot be configured for touch screen applications).

By default, when the operator presses the Cancel button at the terminal, the Cursor Point character becomes invisible for the Numeric Input Cursor Point or Set Bit Cursor Point object. You may want to keep the character visible, to give the operator a visual indication of which Numeric Input Cursor Point or Set Bit Cursor Point is active.

To keep the Cursor Point character visible:

- Click the Retain Cursor On Cancel checkbox to enable it. The Cursor Point character will remain visible, even after the Cancel button is pressed.



Note: If you enable the Retain Cursor On Cancel option, the application will be compatible with PanelView Version 3 or later; it will not be compatible with earlier versions.

For more information about the operation of Set Bit Cursor Points and Numeric Input Cursor Points at the terminal, see “Cursor Point Default Operation” in the chapter *Control Selectors* in the *PanelBuilder 1400e Configuration Software for Windows Reference Manual.*

Configuring PLC I/O Control Options

This chapter explains the PLC I/O Control Options. It provides information about:

- what each PLC I/O Control Option is
- how the PLC I/O Control Options are used
- how to define controls for each PLC I/O Control Option

Configuring PLC I/O Control Options

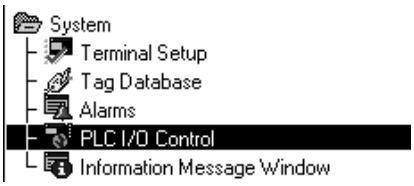
You can instruct the programmable controller to control and monitor certain functions in the terminal by assigning PLC I/O Control Options. For example, to have the programmable controller manage which screen is displayed on the terminal, you must assign a control tag or expression to the PLC Controlled Screen Number option.

Alternatively, the operator at the PanelView terminal can control these options and send the option status to the PLC. The options are then referred to as the Notify PLC options. Using the same example, you would assign a tag to the Screen Number to PLC control.

These three options can be controlled by either the PLC or the operator at the PanelView terminal:

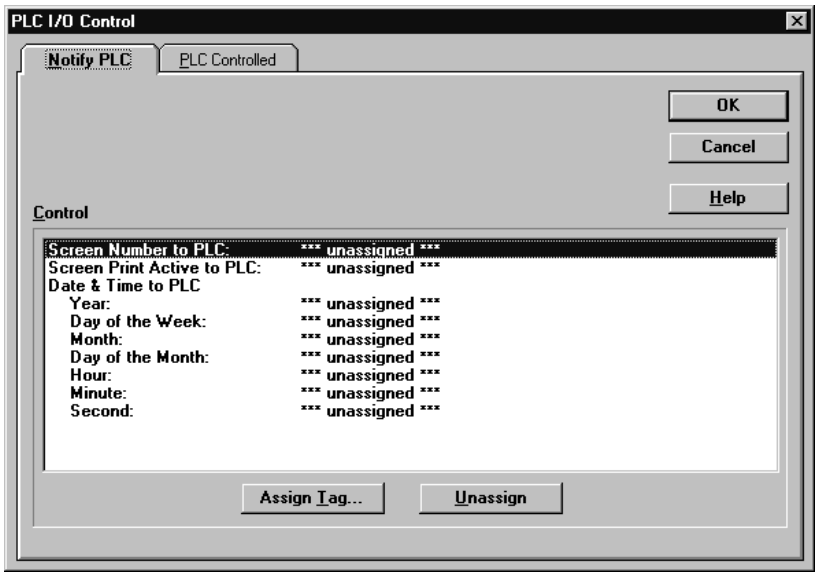
- Screen Number
- Screen Print
- Date and Time

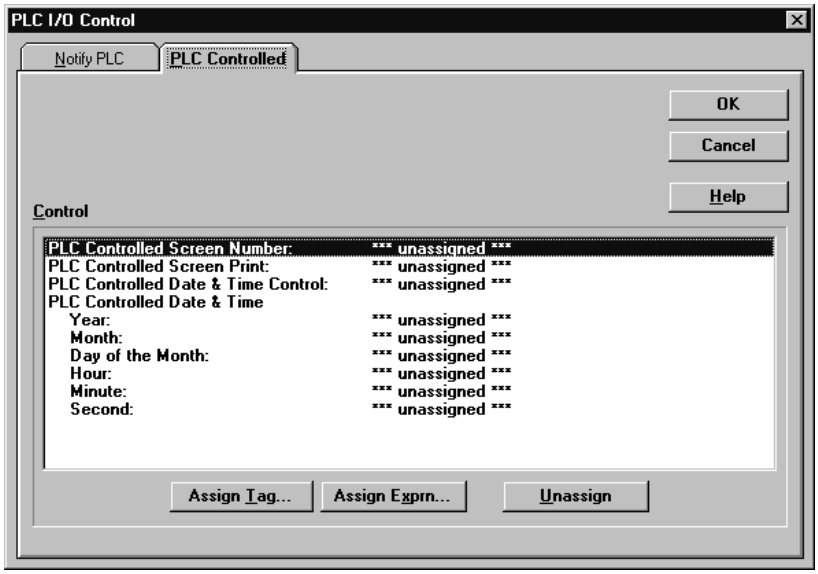
You configure all these PLC I/O Control options in the PLC I/O Control dialog box.



To open the PLC I/O Control dialog box:

- Open the System folder and choose PLC I/O Control, or choose from the toolbar. The following dialog box opens.





To assign or unassign a PLC I/O Control option:

1. In the PLC I/O Control dialog box, select either the Notify PLC tab or the PLC Controlled tab.
2. Highlight the desired control option and choose Assign Tag, Assign Exprn (Expression), or Unassign. The control options are described in the next two sections.

If you are assigning a tag, the Tag Browser dialog box appears. For details on how to assign a tag or create a new one, see Chapter 6, *Defining Tags*.

If you are assigning an expression, the Expression Editor appears. For information about expressions, see Chapter 7, *Creating Expressions*.

If you unassign a tag, the tag remains in the database, but is no longer assigned to the option. If you unassign an expression, it is deleted.

3. Choose OK.

PLC Controlled Options

The PLC Controlled Options include these controls:

- PLC Controlled Screen Number
- PLC Controlled Screen Print
- PLC Controlled Date & Time Control, which works with the PLC Controlled Date & Time tags or expressions.

PLC Controlled Screen Number

With the PLC Controlled Screen Number assigned, the programmable controller can display specific screens. The screen the programmable controller chooses will override the operator's selection and take the operator to the current screen for that state of the operation.

The programmable controller relinquishes screen change control back to the operator when the tag or expression's value is zero.

Important: Ensure your tag address is large enough to accommodate the largest screen number, including the screen numbers of the Alarm History and Alarm Status screens.

PLC Controlled Screen Print

By activating the PLC Controlled Screen Print option, you allow the programmable controller to trigger a screen print of the currently active screen and pop-up windows such as the Alarm and Information Windows.

When the tag or expression value changes from 0 to non-zero, the screen and pop-up windows displayed on the terminal will be printed. The programmable controller must maintain the set bit long enough to ensure that the terminal doesn't miss it.

Note: You can also allow manual screen printing by adding a Screen Print Button to screens. Refer to Chapter 9, *Creating Objects*, for more information on creating buttons.

About Screen Prints from the Terminal

Screen prints can be requested by the programmable controller, or initiated manually by the operator pressing a Screen Print Button object.

If a PLC Controlled Screen Print is detected in the same Remote I/O scan as a PLC Controlled Screen Change, the screen will change before the new screen is printed. Screen prints are depicted as they actually are, but in gray-scale rather than color.

Important: A screen print captures the terminal screen image and prints a gray-scale representation of the screen. If you are using a PLC Controlled Screen Print option, ensure that your screen saver is disabled. Otherwise, if the screen saver is displayed when the print request is received, only the screen saver, not the application screen and pop-up windows, will be printed.

PLC Controlled Date & Time Control

Assigning a digital tag to the PLC Controlled Date & Time Control allows the programmable controller to obtain the time and date and to reset the terminal's internal clock to match. The PLC Controlled Date & Time Control defines the PLC address that triggers the terminal to get the time and date from the individual PLC Controlled Date & Time controls.

This option is especially useful if you have several terminals connected to the same programmable controller. Instead of an operator having to reset the date and time on each terminal, the programmable controller can reset all internal clocks at once.

When the PLC Controlled Date & Time Control value changes from 0 to non-zero, the terminal resets its internal time and date to the time and date read from the individual PLC Controlled Date & Time controls.

Important: You must also configure the individual PLC Controlled Date & Time controls (described in the next section) for the PLC Controlled Date & Time Control to function.

Individual PLC Controlled Date & Time Controls

The individual PLC Controlled Date & Time controls work with the PLC Controlled Date & Time Control. The PLC Controlled Date & Time controls define the programmable controller addresses from which the terminal gets the time and date.

There are six controls:

This control:	Accepts this range of data:
Year 2-digit form 4-digit form	 80 – 99 (1980 – 1999) and 00 – 43 (2000 – 2043) 1980 – 2043
Month	1 – 12
Day of the Month	1 – 31
Hour	0 – 23 (24 hour format)
Minute	0 – 59
Second	0 – 59

These controls are all individually assignable independent of the others. Only the date and time components for which tags or expressions are assigned will be updated. For example, if only the Year control is assigned, then the programmable controller will only be asked to update the year, not the month, day of the month, or time.

The way these controls display the date and time on the PanelView terminal depends on the terminal's Configure Mode settings for the date and time display. For information about configuring the PanelView terminal to display the date and time, see "Setting Time and Date" in Chapter 5, *Configuring PanelView Terminals*, in the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual*.

If an error is detected in the data received from any of the assigned controls while the application is running, the data is discarded and a minor fault message is displayed.

Notify PLC Options

Options that are passed from the terminal to the programmable controller include these controls:

- Screen Number to PLC
- Screen Print Active to PLC
- Date & Time to PLC

Screen Number to PLC

Assign the Screen Number to PLC control to transfer the current screen number from the terminal to the programmable controller.

This option is useful for multiplexing display data from programmable controller output addresses based on screen number.

Important: Ensure your tag address is large enough to accommodate the largest screen number, including the screen numbers of the Alarm History and Alarm Status screens.

Screen Print Active to PLC

If you try to make screen prints while running an application on the terminal, but the printer is busy, it will ignore your screen print request.

If you assign the Screen Print Active to PLC control, the terminal informs the programmable controller that a screen print is in progress.

A tag value of 1 indicates the Screen Print Active condition; a tag value of 0 indicates the non-active condition, and that the printer is free.

Date & Time to PLC

Use these options if you want the programmable controller to store the current time and date from the terminal's clock.

You can use 7 separate controls, which are listed in the table below. Any of them can be assigned or left unassigned. When the value of any one of the assigned controls changes, the values of all the assigned controls are updated.

Note: For Remote I/O applications, all assigned tags are sent to the PLC in the same programmable controller scan. However, for DH+ and ControlNet applications, it is not possible to guarantee synchronicity. The controls are updated in the order in which they appear in the following table:

This control:	Sends this range of data to the PLC:
Year (2-digit form)	0 – 99
Day of the Week	1 – 7 (1=Sunday)
Month	1 – 12
Day of the Month	1 – 31
Hour	0 – 23 (24-hour format)
Minute	0 – 59
Second	0 – 59

Working with the Information Message Window

This chapter explains:

- what the Information Message Window is and how it works at runtime
- how to configure the Information Message Window in PanelBuilder 1400e

About the Information Message Window

The programmable controller triggers the Information Message Window to display messages when certain events or processes occur. The messages that appear in the Information Message Window are the information messages you define in the Information Messages editor in PanelBuilder.

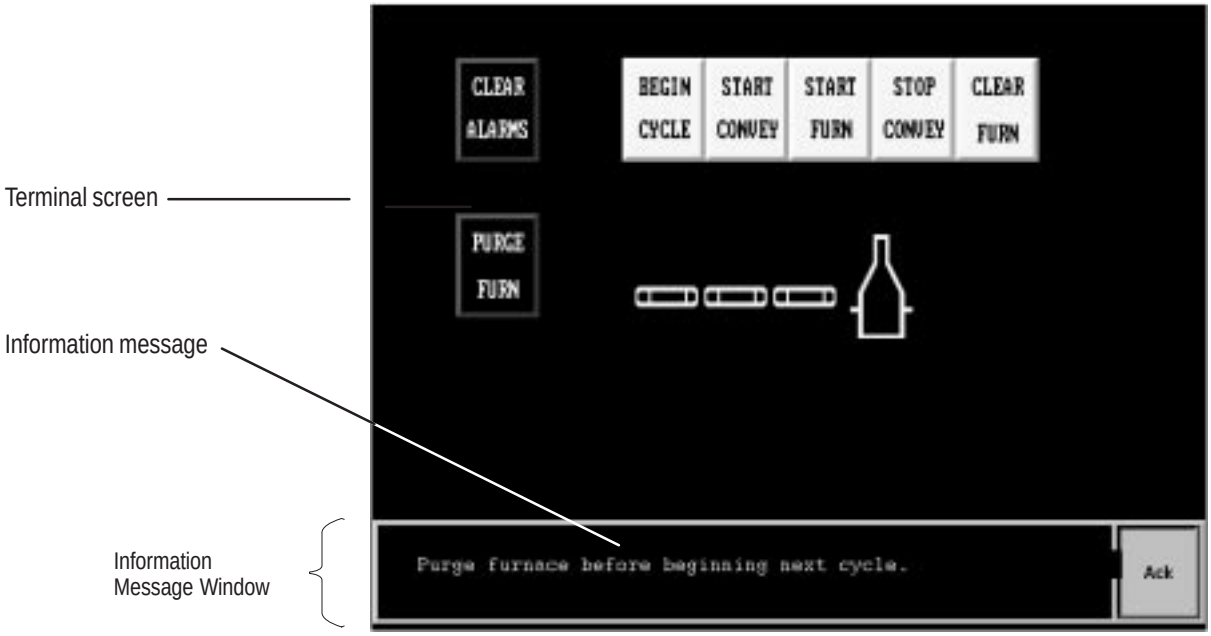
Note: You create the information messages in the Information Messages editor, not in the Information Message Window. For details about creating information messages, see Chapter 10, *Creating Messages.*

The Information Message Window provides the operator with informative messages about the process, prompts or instructions, and information about current states.

When the application is running on the terminal, the Information Message Window appears over the currently-displayed application screen. It is not affected when application screens change.

Using the Information Message Window in the Terminal

The Information Message Window appears at the bottom of the screen on the terminal.



To trigger an information message, the programmable controller places a non-zero value in the Information Window Control. This value must correspond to the trigger value of a message you created in the Information Messages editor. The terminal reads this trigger value and displays the corresponding information message.

Depending on how you configure the Information Message Window, the window will remain on the screen until one of the following occurs:

- The programmable controller sets the value at the Information Window Control to 0.
- The operator presses the window's **Clear** button.
- The operator presses the window's **Ack** button. (This button appears only if the Acknowledge to PLC control has been enabled.)

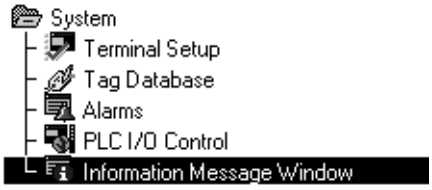
  When the operator presses the **Ack** button, the Acknowledge to PLC tag is set to 1 for the duration of the Information Window Ack Hold Time. The PLC can be programmed to instruct the terminal to clear the window by setting the Information Window Control back to zero.

Note: If the Information Message Window is on-screen at the moment when an alarm occurs, the Alarm Window may overlap the Information Message Window. The Information Message Window will remain after the Alarm Window is cleared.

Configuring the Information Message Window

When you configure the Information Message Window, you can:

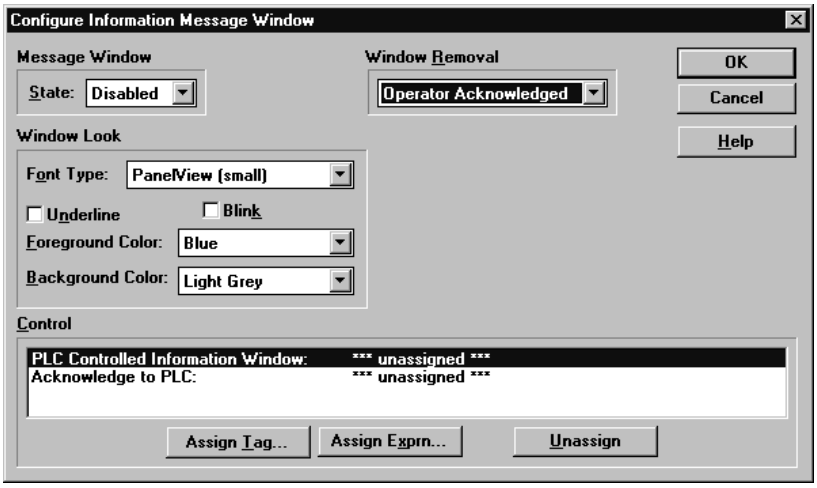
- enable or disable it
- specify how the window is removed from the terminal's screen
- change the default colors and other aspects of the window's appearance
- define the control that the PLC uses to trigger the Information Message Window
- define the Acknowledge to PLC control if you specify that the operator must acknowledge each message



To configure the Information Message Window:

1. Open the System folder and choose Information Message Window, or choose from the toolbar.

   The Configure Information Message Window dialog box appears.



2. Configure the window to your specifications by filling in each field as follows:

   Message Window State
   The Information Message Window is an optional feature in your application. By default it is disabled. To make it appear, select Enabled in the State drop-down list. If you don't want it to appear, leave it disabled.

   ▶ **Note:** Information messages defined in the Information Messages editor will not appear if the window is disabled.

Window Removal
At runtime, the Information Message Window can be removed from the terminal's screen in one of three ways. Select the desired option from the Window Removal drop-down list:

- **PLC Cleared**—the programmable controller clears the window by setting the value at the Information Window Control to 0. If you select this option the Information Message window will not contain a Clear or Acknowledge button.
- **Operator Cleared**—the operator presses the window's **Clear** button and the terminal removes the window.
- **Operator Acknowledged**—the operator presses the window's **Ack** button. The terminal then sets the Acknowledge to PLC control to a value of 1. The programmable controller must be programmed to monitor this tag. When the tag is set to 1, the programmable controller sets the Information Window Control to 0. This removes the window. The Acknowledge to PLC control is set to 1 for the Information Window Ack Hold Time.

Font Type
You can choose from four different font sizes: Small, Double High, Double Wide, and Large. If you are creating a Version 2 application, each message can contain up to 144 characters.

For earlier versions, the font size you choose determines the maximum length of the messages:

- If you choose Small or Double High, the messages can be up to 70 characters.
- If you choose Double Wide or Large, the message can be up to 35 characters.

Blink
Click the check box to choose Yes, if the messages are to blink. Leave it unchecked to choose No.

Underline
Click the check box to choose Yes, if the messages are to be underlined. Leave it unchecked to choose No.

Foreground Color
Choose the foreground color for the text in the Information Message Window.

Background Color
Choose the background color for the Information Message Window.

3. Assign the Information Window Control, and if appropriate, the Acknowledge to PLC control. These two controls are explained in more detail in the following table.

This control:	Does this:
PLC Controlled Information Window	Directs the terminal to display a message. The programmable controller places a trigger value at this control's address, and the terminal displays the message with the corresponding trigger value. You must assign this control for the Information Message Window to function.
Acknowledge to PLC	Notifies the PLC when the ACK button is pressed. This enables the programmable controller to control the window display until the operator acknowledges it. You must assign this control if you've specified Operator Acknowledged as the window removal method.

For details on assigning and creating tags, see Chapter 6, *Defining Tags*. For details on assigning and creating expressions, see Chapter 7, *Creating Expressions*.

4. When you're finished configuring the Information Message Window, choose OK to accept the changes and close the dialog box.

Transferring Applications

About Transferring Applications

To transfer application files to or from PanelView terminals, the PanelView 1200/1400e Transfer Utility Version 3 or PanelView e FTU 32 Version 4 or later, must be installed. These utilities are included with your PanelBuilder software kit, and are also available separately as Catalog Number 2711E-ND7.

For information about installing the PanelView 1200/1400e Transfer Utility or PanelView e FTU 32, and transferring application files, see the *PanelView 1200e/1400e Transfer Utility User Manual*, Publication Number 2711E-6.8.

For specific information about transferring Modbus application files, see the *PanelBuilder 1400e Configuration Software for Windows Modbus User Manual*, Publication Number 2711E-6.12.

There are two ways to transfer an application from Panelbuilder to a PanelView terminal:

- from within PanelBuilder, if you have opened the application file and are working on it, or
- using the PanelView 1200/1400e Transfer Utility or PanelView e FTU 32, if you have already created a .pvd (PanelView terminal) file

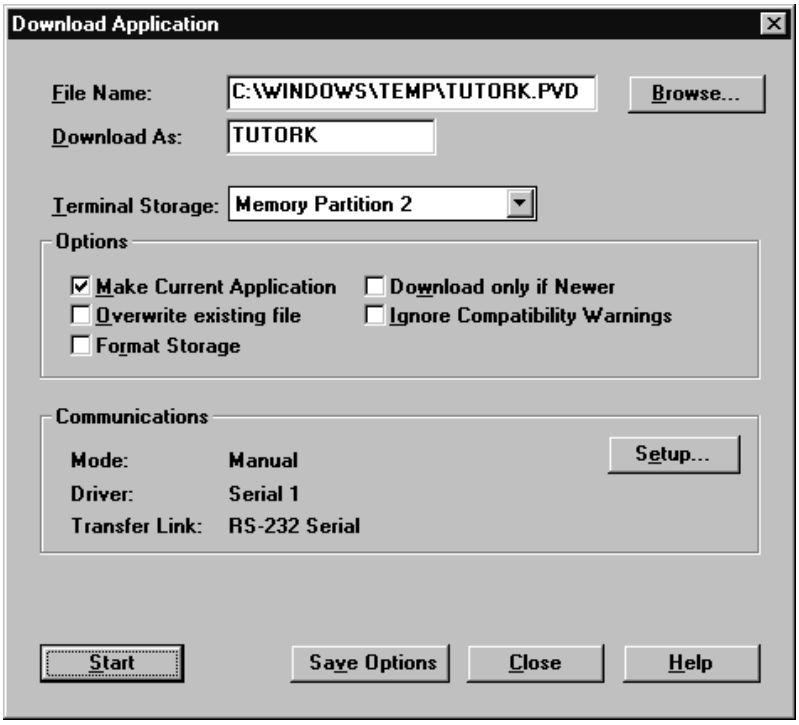
Whichever method you choose, the process of configuring the communication parameters and upload or download options is the same.

To transfer an application from within PanelBuilder:

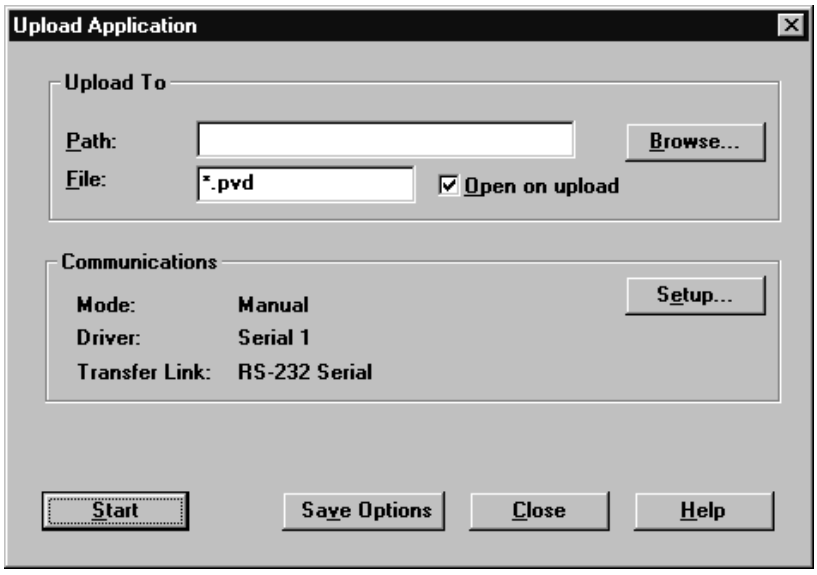
PanelBuilder transfers applications using the Transfer Utility. However, for your convenience, you can configure and initiate the upload or download without leaving PanelBuilder.

1. Prepare the PanelView terminal to receive the upload or download. For details see Chapter 3, *Uploading 1000e/1200e/1400e Application Files and Alarm History,* and Chapter 4, *Downloading 1000e/1200e/1400e Application Files*, in the *PanelView 1200/1400e Transfer Utility User Manual* (Publication Number 2711E-6.8).

2. Choose Download from the File menu, or from the toolbar,



or choose Upload from the File menu or from the toolbar.



3. For information about configuring the upload or download once the Upload Application or Download Application dialog box appears, see Chapter 3, *Uploading 1000e/1200e/1400e Application Files and Alarm History,* Chapter 4, *Downloading 1000e/1200e/1400e Application Files,* and Chapter 5, *Configuring Communications Settings for 1000e/1200e/1400e Application Transfers,* in the *PanelView 1200/1400e Transfer Utility User Manual* (Publication Number 2711E-6.8).

To transfer an application from the PanelView 1200/1400e Transfer Utility:

For information about transferring applications using the Transfer Utility, see the *PanelView 1200e/1400e Transfer Utility User Manual (*Publication Number 2711E-6.8).

Sample Worksheets

The worksheets in this appendix are designed to help you plan your application. To complete these worksheets, see Chapter 2, *Planning Applications*. Make copies of the worksheets as you need them and save the originals for future use.

This Appendix contains the following worksheets:

- Discrete I/O Usage Worksheet

  Use this worksheet to help you keep track of the PLC addresses. Use one for each full or partial rack assignment.
- Block Transfer File I/O Usage Worksheet

  Use this worksheet to help you keep track of the block transfer addresses. Use one worksheet for each block transfer file.
- Rack Assignments Worksheet

  Use this worksheet to record your exact rack assignments.
- Block Transfer File Worksheet

  Use this worksheet to record your block transfer file assignments.
- Remote I/O Tags Worksheet

  Use this worksheet to help you assign programmable controller addresses to each object on a screen.
- Communications Worksheet:
  - DH+ Communications Worksheet
  - ControlNet Communications Worksheet

  Use these worksheets to record details of the programmable controllers the terminal will be communicating with.
- Device Tags Worksheet

  Use this worksheet to help you assign programmable controller addresses to each object on a screen.
- Screen Worksheet for 1200e Terminals

  Use this worksheet to plan the object positions for each screen you want to create for PanelView 1200e Keypad or Touch Screen terminal applications.
- Screen Worksheet for 1400e Terminals

  Use this worksheet to plan the object positions for each screen you want to create for PanelView 1400e Keypad or Touch Screen terminal applications.
- Screen Worksheet for 1000e Terminals

  Use this worksheet to plan the object positions for each screen you want to create for PanelView 1000e Keypad or Touch Screen terminal applications.

Discrete I/O Usage Worksheet

Application File Name: _____ Rack Number: _____

Full or Listen Only Access (Circle One)

PLC Output Bit Numbers

	17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0
0																
1																
2																
3																
4																
5																
6																
7																

PLC Input Bit Numbers

	17	16	15	14	13	12	11	10	7	6	5	4	3	2	1	0
0																
1																
2																
3																
4																
5																
6																
7																

Block Transfer File I/O Usage Worksheet (Page 1 of 2)

Application File Name: _____

Block Transfer File Number: _____ (Enter a number between 1 and 64)

Number of Words: _____ (Enter a number between 1 and 64)

READ (PLC In) or **WRITE** (PLC Out) (Circle One)

FULL or **LISTEN ONLY** Access (Circle One)

PLC Bit Numbers

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0																
1																
2																
3																
4																
5																
6																
7																
8																
9																
10																
11																
12																
13																
14																
15																
16																
17																
18																
19																
20																
21																
22																
23																
24																
25																
26																
27																
28																
29																
30																
31																

Block Transfer File I/O Usage Worksheet (Page 2 of 2)

Application File Name: _____

Block Transfer File Number: _____ (Enter a number between 1 and 64)

Number of Words: _____ (Enter a number between 1 and 64)

READ (PLC In) or **WRITE** (PLC Out) (Circle One)

FULL or **LISTEN ONLY** Access (Circle One)

PLC Bit Numbers

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
32																
33																
34																
35																
36																
37																
38																
39																
40																
41																
42																
43																
44																
45																
46																
47																
48																
49																
50																
51																
52																
53																
54																
55																
56																
57																
58																
59																
60																
61																
62																
63																

Rack Assignments Worksheet

Application File Name: _____

PLC/Scanner Type: _____

Baud Rate: 57.6K (10,000 Ft) 115.2K (5,000 Ft) 230.4K (2,500 Ft)

Note: You must make at least one rack assignment. Additional rack assignments are optional. Record the Rack Number and circle the Module Groups, Last Chassis designation, and Access used. Use additional worksheets if necessary.

Rack Assignments

Rack Number	Module Groups				Last Chassis		Access	
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only
	0,1	2,3	4,5	6,7	Y	N	Full	Listen Only

Block Transfer File Worksheet

Application File Name: _____

PLC/Scanner Type: _____

Baud Rate: 57.6K (10,000 Ft) 115.2K (5,000 Ft) 230.4K (2,500 Ft)

Notes:

1. Block Transfers are optional. Record the Block Transfer File Number and Size, write in one of the Rack numbers from the previous worksheet, circle the appropriate Word and Byte, circle Read or Write, and Access used. Use additional worksheets if necessary.
2. If the Block Transfer File is on a Listen Only rack, the file can only have Listen Only access.
3. The Pass Through Block Transfer File must be defined on a Full Access rack.
4. For each block transfer file, mark off the byte in the output image on the Discrete I/O Usage Worksheet.

Block Transfer File Assignments

Block Transfer File Number	Size	Rack	Word	Byte	Usage	Access
Pass Through	64		0 1 2 3 4 5 6 7	Hi Lo	Both	Full
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only
			0 1 2 3 4 5 6 7	Hi Lo	Read Write	Full Listen Only

Remote I/O Tags Worksheet

Application File Name: _____ **Screen Number:** _____

Object			Tag							Address		
Ref No.	Type	Initial State Value	Name	Type	Analog Tag only			Initial Value	Range of Values	Start Address	No. of Bits, End Address or (string tag only) No. of characters	Tag Start Value
					Scale	Offset	Data Type					

Address formats:

Discrete:	I/O:rrw/bb	I is input, O is output; rr is rack number; w is start word; /bb is optional bit offset. All numbers are octal.	e.g., I:011/12
Block Transfer:	Bff ww/bb	B is block transfer; ff (followed by a space) is file number; ww is start word; /bb is optional bit offset. All numbers are octal.	e.g., B1 01/01
Discrete SLC:	I/O:ss.ww/bb	I is input, O is output; ss is optional slot number; ww is start word; /bb is optional bit offset. All numbers are decimal.	e.g., O:01.18

Notes:

Initial State Value: Objects such as Multistate Push Buttons have a user-configurable initial state whose value will supersede the tag's initial value when the application is first run. This is called the tag's Start Value.

Tag Start Value: The start value is the tag initial value, or the object initial state value, if one exists.

DH+ Communications Worksheet

Application File Name: _____

PanelView Station Number: _____

Baud Rate: 57.6K (10,000 Ft)

Node Definitions

Node Name	PLC Type	Station Address	Timeout (sec)
Unsolicited_Msgs	PLC–2	*	3.000

***** The Unsolicited Messages Node is the PanelView terminal. By default, the terminal's Station Address is the DH+ station number configured in the current application. The Station Address can be changed in the PanelView terminal's Terminal Network Setup screen.

Note: For more node definitions, use additional copies of this worksheet.

Scan Class Definitions

Name	Foreground (sec)	Background (sec)	Name	Foreground (sec)	Background (sec)

Note: The default foreground and background rates are 0.000 seconds and 5.000 seconds, respectively.

ControlNet Communications Worksheet

Application File Name: _____

PanelView Station Number: _____

Node Definitions

Node Name	PLC Type	Station Address	Timeout (sec)
Unsolicited_Msgs	PLC–2	*	3.000

***** The Unsolicited Messages Node is the PanelView terminal. By default, the terminal's Station Address is the station number configured in the current application. The Station Address can be changed in the PanelView terminal's Terminal Network Setup screen.

Note: For more node definitions, use additional copies of this worksheet.

Scan Class Definitions

Name	Foreground (sec)	Background (sec)	Name	Foreground (sec)	Background (sec)

Note: The default foreground and background rates are 0.000 seconds and 5.000 seconds, respectively.

Device Tags Worksheet

Application File Name: _____ **Screen Number:** _____

Object			Tag							Address		
Ref No.	Type	Initial State Value	Name	Type	Analog Tag Only			Initial Value	Range of Values	Address	Number of characters (string tag only)	Tag Start Value
					Scale	Offset	Data Type					

Notes:

Initial State Value: Objects such as Multistate Push Buttons have a user-configurable initial state whose value will supersede the tag's initial value when the application is first run. This is called the tag's start value.

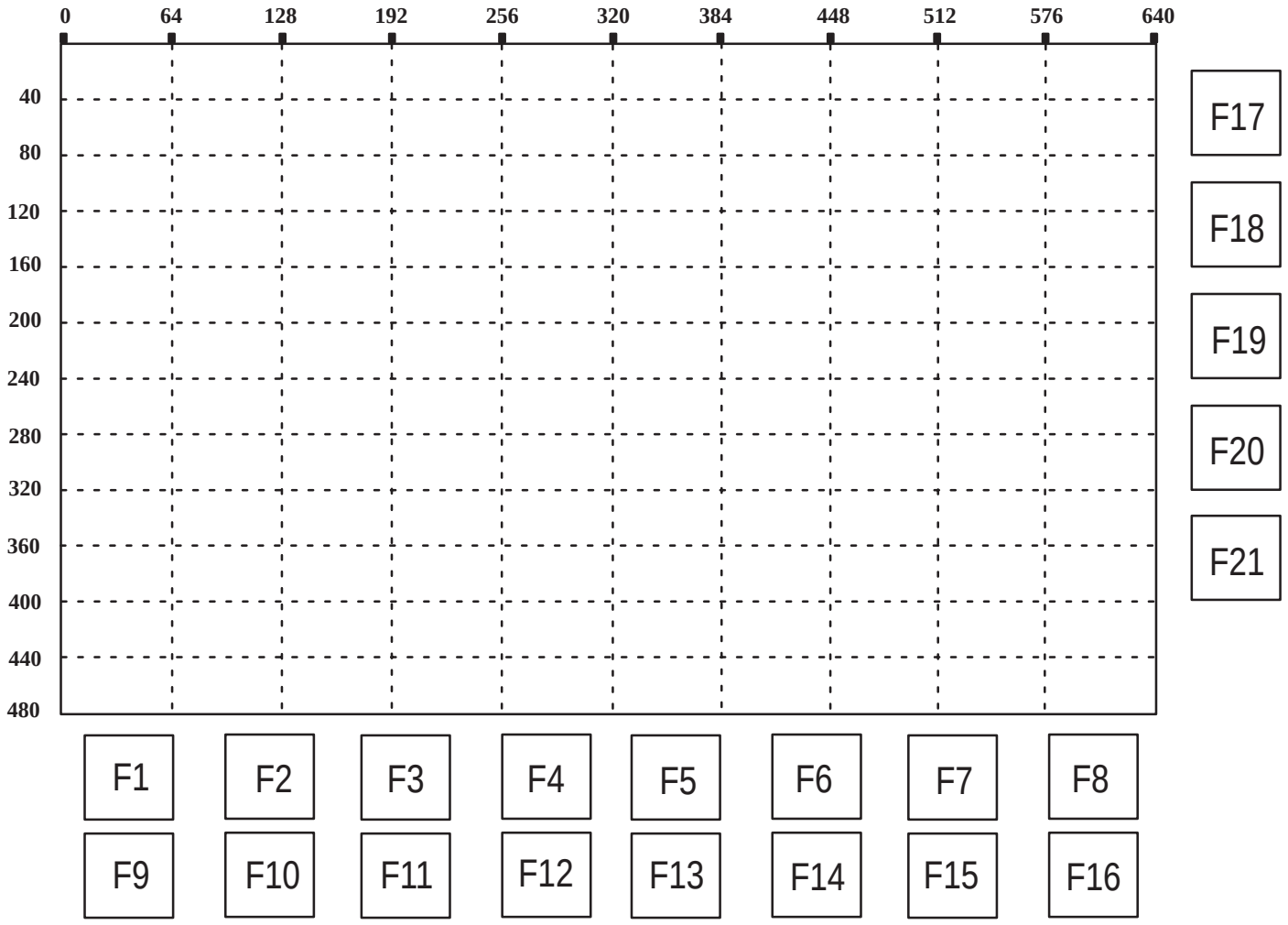
Tag Start Value: The start value is the tag initial value, or the object initial state value, if one exists.

For information about addressing syntax, refer to Appendix C, *Addressing Syntax.*

Screen Worksheet for 1200e Terminals

Application File Name: _____

Screen Number: _____ **Screen Name:** _____



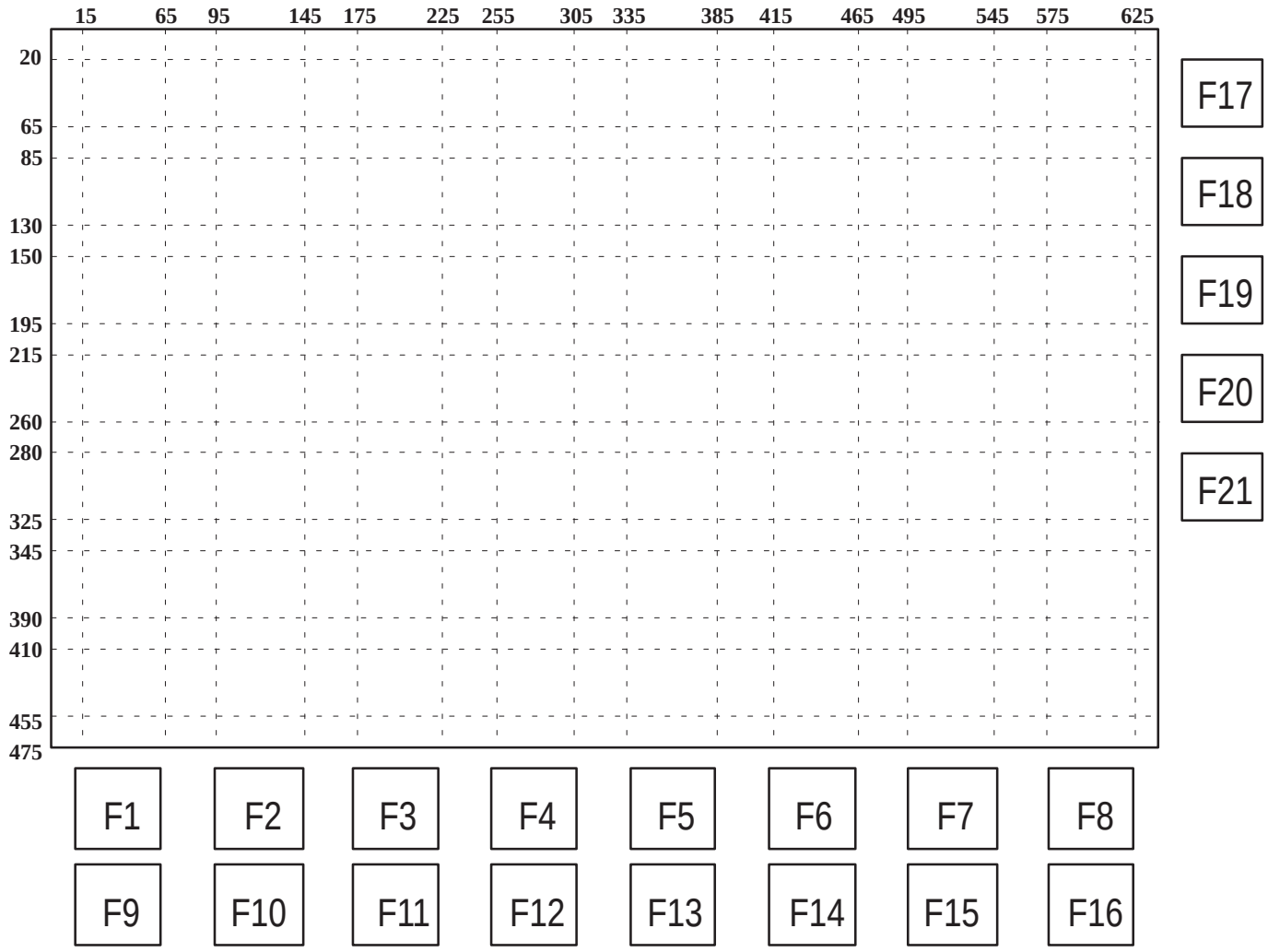
Notes:

1. The grid shows the location of the touch cells on a PanelView 1200e Touch Screen terminal. PanelBuilder sizes buttons in multiples of touch cells. The smallest size for a 1200e touch screen application input object is 64 by 40 pixels.
2. You can use this worksheet to align buttons you draw on the screen to keypad terminal function keys.

Screen Worksheet for 1400e Terminals

Application File Name: _____

Screen Number: _____ **Screen Name:** _____



Notes:

1. You can use this worksheet to align buttons you draw on the screen to keypad terminal function keys.

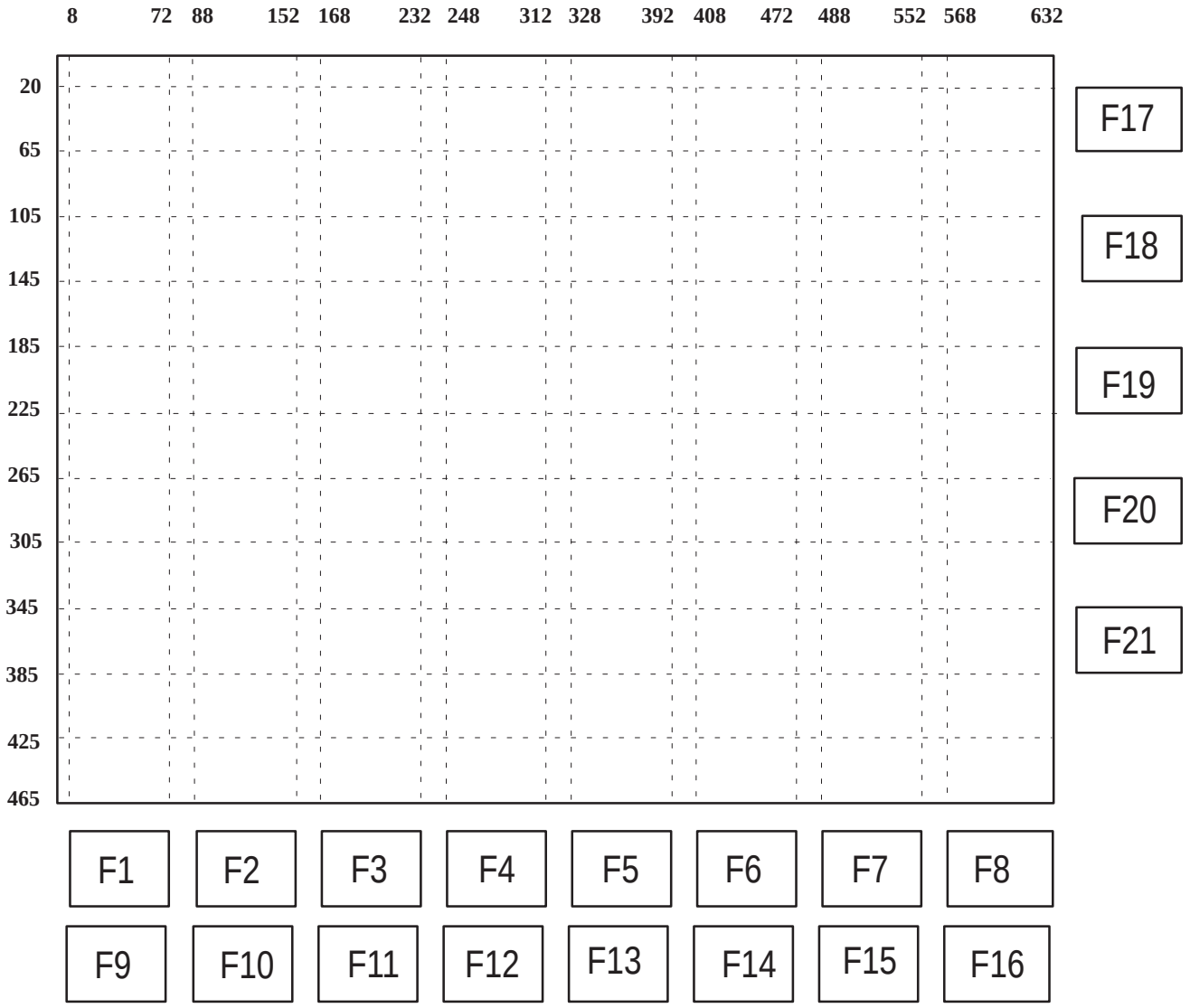
2. The PanelView 1400e touch screen terminals use analog-resistive touch screens. This allows input objects to be any size (minimum 40 by 40 pixels) and to be placed anywhere on the screen. The analog-resistive touch screen provides greater flexibility in screen design compared to PanelView 1200 and 1200e terminals. PanelView 1200 and 1200e terminals use a touch matrix which requires that all input objects align with a touch grid.

ATTENTION: Do not press multiple touch screen locations at the same time. In the unlikely event that two or more touches occur simultaneously in the same horizontal or vertical line, an input object located half-way between the two touches could be activated. This is inherent in the analog-resistive touch screen technology used in PanelView 1400e terminals.

Screen Worksheet for 1000e Terminals

Application File Name: _____

Screen Number: _____ **Screen Name:** _____



Notes:

1. You can use this worksheet to align buttons you draw on the screen to keypad terminal function keys.

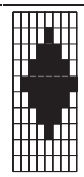
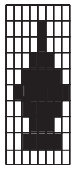
2. The PanelView 1000e touch screen terminals use analog-resistive touch screens. This allows input objects to be any size (minimum 40 by 40 pixels) and to be placed anywhere on the screen. The analog-resistive touch screen provides greater flexibility in screen design compared to PanelView 1200 and 1200e terminals. PanelView 1200 and 1200e terminals use a touch matrix which requires that all input objects align with a touch grid.

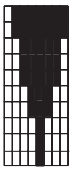
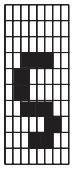
ATTENTION: Do not press multiple touch screen locations at the same time. In the unlikely event that two or more touches occur simultaneously in the same horizontal or vertical line, an input object located half-way between the two touches could be activated. This is inherent in the analog-resistive touch screen technology used in PanelView 1000e terminals.

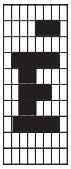
The Extended Character Set

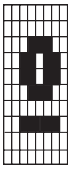
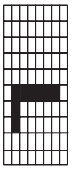
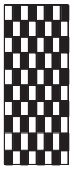
You can use PanelBuilder to create characters not on your keyboard. The following tables list the extended characters available in PanelBuilder.

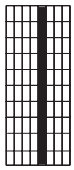
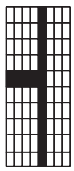
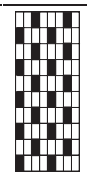
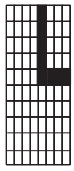
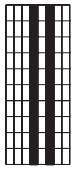
Hold down the ALT key and type the specified number on the numeric keypad.

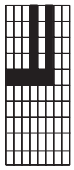
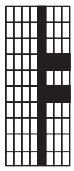
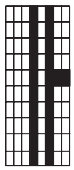
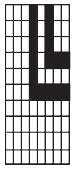
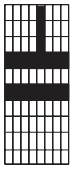
Key	Character	Key	Character
ALT-01		ALT-02	
ALT-04		ALT-05	
ALT-06		ALT-07	
ALT-010		ALT-011	
ALT-012		ALT-014	
ALT-015		ALT-016	
ALT-017		ALT-018	

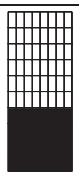
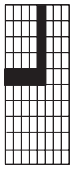
Key	Character	Key	Character
ALT-019		ALT-020	
ALT-021		ALT-023	
ALT-024		ALT-025	
ALT-026		ALT-029	
ALT-030		ALT-031	
ALT-0128		ALT-0135	
ALT-0129		ALT-0136	
ALT-0130		ALT-0137	
ALT-0131		ALT-0138	

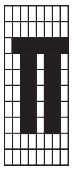
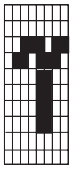
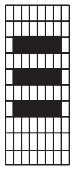
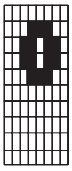
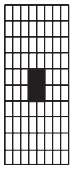
Key	Character	Key	Character
ALT-0132		ALT-0139	
ALT-0133		ALT-0140	
ALT-0134		ALT-0141	
ALT-0142		ALT-0149	
ALT-0143		ALT-0150	
ALT-0144		ALT-0151	
ALT-0145		ALT-0152	
ALT-0146		ALT-0153	
ALT-0147		ALT-0154	

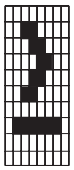
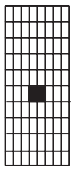
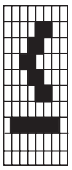
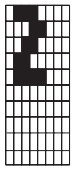
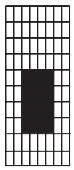
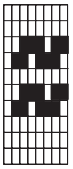
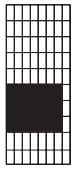
Key	Character	Key	Character
ALT-0148		ALT-0155	
ALT-0156		ALT-0163	
ALT-0157		ALT-0164	
ALT-0158		ALT-0165	
ALT-0159		ALT-0166	
ALT-0160		ALT-0167	
ALT-0161		ALT-0168	
ALT-0162		ALT-0169	
ALT-0170		ALT-0177	

Key	Character	Key	Character
ALT-0171		ALT-0178	
ALT-0172		ALT-0179	
ALT-0173		ALT-0180	
ALT-0174		ALT-0181	
ALT-0175		ALT-0182	
ALT-0176		ALT-0183	
ALT-0184		ALT-0191	
ALT-0185		ALT-0192	
ALT-0186		ALT-0193	

Key	Character	Key	Character
ALT-0187		ALT-0194	
ALT-0188		ALT-0195	
ALT-0189		ALT-0196	
ALT-0190		ALT-0197	
ALT-0198		ALT-0205	
ALT-0199		ALT-0206	
ALT-0200		ALT-0207	
ALT-0201		ALT-0208	
ALT-0202		ALT-0209	

Key	Character	Key	Character
ALT-0203		ALT-0210	
ALT-0204		ALT-0211	
ALT-0212		ALT-0219	
ALT-0213		ALT-0220	
ALT-0214		ALT-0221	
ALT-0215		ALT-0222	
ALT-0216		ALT-0223	
ALT-0217		ALT-0224	
ALT-0218		ALT-0225	

Key	Character	Key	Character
ALT-0226		ALT-0233	
ALT-0227		ALT-0234	
ALT-0228		ALT-0235	
ALT-0229		ALT-0236	
ALT-0230		ALT-0237	
ALT-0231		ALT-0238	
ALT-0232		ALT-0239	
ALT-0240		ALT-0248	
ALT-0241		ALT-0249	

Key	Character	Key	Character
ALT-0242		ALT-0250	
ALT-0243		ALT-0251	
ALT-0244		ALT-0252	
ALT-0245		ALT-0253	
ALT-0246		ALT-0254	
ALT-0247		ALT-0255	

Addressing Syntax

This appendix describes the addressing syntax for:

- PLCs
- SLCs
- Remote I/O

For details more specific to your PLC, consult the User's Manual for your PLC.

PLC Addressing Syntax

The following section provides addressing syntax for the following types of programmable controllers:

- PLC-2
- PLC-3
- PLC-5, which includes the PLC-5/10, PLC-5/12, PLC-5/15, and PLC-5/25
- PLC-5 (Enhanced), which includes the PLC-5/11, PLC-5/20, PLC-5/20C, PLC-5/30, PLC-5/40, PLC-5/40C, PLC-5/60, and PLC-5/80

PLC-2

aaaa/bb

aaaa	Address: 0 – 7777 octal
bb *(optional)*	Bit offset within word: 0 – 17 octal

Example of digital address: 11/17
Example of analog and string address: 1264

PLC-3 Data Table Sections

Fnnn:wwww/bb

F	File type: A = ASCII B = Binary N = Integer D = Decimal (BCD) O = Output I = Input F = Floating point S = Status H = High order integer
nnn	File number: 0 – 999 decimal
wwww	Word address: For I/O section: 0 – 7777 octal For all others: 0 – 9999 decimal
bb *(optional)*	Bit offset within word: 0 – 17 octal Bit offset not permitted in the H and F sections.

Example of digital address: B5:173/15
Example of analog address: N3:173
Example of string address: A1:126

PLC-3 Timers and Counters

FWWW:nnnn/bb

F	File type: C = Counter T = Timer
WWW	Structured word: CTL = Control word PRE = Preset value ACC = Accumulated value
nnnn	Counter/Timer number: 0 – 9999 decimal
bb	Bit offset within word: 15 octal = done bit 16 octal = timing bit 17 octal = enable bit

Example of digital address: TCTL:125/17
Example of analog address: CPRE:125

PLC-5 Data Files

Fnnn:eee/bb

F	File type: B = Binary N = Integer D = Decimal (BCD) F = Floating point A = ASCII
nnn	File number: 3 – 999 decimal Binary, integer, and floating point file types use the default file number if the file number is absent. The default numbers are 3 (binary), 7 (integer), and 8 (floating point).
eee	Element number: 0 – 999 decimal
bb *(optional)*	Bit offset within word: 0 – 15 decimal Not applicable to floating point file types.

Example of bit address: B3:173/15
Example of word address: N7:12

PLC-5 Binary Files: Optional Syntax

Fnnn/bbbbb

F	File type: B = Binary
nnn	File number: 3 – 999 decimal Uses default file number 3 if the file number is absent.
bbbbb	Bit offset from start of file: 0 – 15999 decimal

Example: B3/1024

PLC-5 I/O and Status Files

F:ee/bb

F	File type: I = Input O = Output S = Status
ee	Element number: For input and output files: PLC-5/10: 0 – 037 octal PLC-5/12: 0 – 037 octal PLC-5/15: 0 – 037 octal PLC-5/25: 0 – 077 octal For status files: 0 – 31 decimal
bb *(optional)*	Bit offset within word: For input and output files: 0 – 17 octal For status files: 0 – 15 decimal

Example: O:64/17

PLC-5 Timers, Counters, and Control Files

Fnnn:eee.MNE/bb

F	File type: C = Counter T = Timer R = Control
nnn	File number: 3 – 999 decimal Timer, counter, and control file types use the default file number if the file number is absent. The default numbers are 4 (timer), 5 (counter), and 6 (control).
eee	Element number: 0 – 999 decimal
MNE	Member mnemonic (see mnemonic tables on Page C–11)
bb *(optional)*	Bit number: 0 – 15 decimal Only applies to analog word members.

Example: T57:38.ACC

PLC-5 (Enhanced) I/O and Status Files

F:eee/bb

F	File type: I = Input O = Output S = Status
eee	Element number: For input and output files: PLC-5/11: 0 – 037 octal PLC-5/20: 0 – 037 octal PLC-5/30: 0 – 077 octal PLC-5/40: 0 – 177 octal PLC-5/60: 0 – 277 octal PLC-5/80: 0 – 277 octal For status files: 0 – 127 decimal
bb *(optional)*	Bit offset within word: For input and output files: 0 – 17 octal For status files: 0 – 15 decimal

Example: O:167/11

PLC-5 (Enhanced) Binary, Integer, BCD, ASCII, Float, and String Files

Fnnn:eee/bb

F	File type: B = Binary D = Decimal (BCD) N = Integer F = Floating point A = ASCII ST = String
nnn	File number: 3 – 999 decimal Binary, integer, and floating point file types use the default file number if the file number is absent. The default numbers are 3 (binary), 7 (integer), and 8 (floating point).
eee	Element number: 0 – 999 decimal

bb *(optional)*	Bit offset within word: 0 – 15 decimal Bit offset is not supported for floating point and string file types.

Example: B84:113/13

PLC-5 (Enhanced) Binary Files: Optional Syntax

Fnnn/bbbbb

F	File type: B = Binary
nnn *(optional)*	File number: 3 – 999 decimal Uses default file number 3 if the file number is absent.
bbbbb	Bit offset from start of file: 0 – 15999 decimal

Example: B27/248

PLC-5 (Enhanced) Timer, Counter, Control, SFC Status, Message, PID, Block Transfer, and Token Data Files

FFnnn:eee.MNE[ss]/bb

FF	File type: T = Timer C = Counter R = Control SC = SFC status MG = Message PD = PID control BT = Block transfer TD = Token data
nnn	File number: 3 – 999 decimal Timer, counter, and control file types use the default file number if the file number is absent. The default numbers are 4 (timer), 5 (counter), and 6 (control).
eee	Element number: 0 – 999 decimal
MNE	Member mnemonic (see mnemonic tables on Page C–11)

ss	Submember Only applies to .ADDR and .DATA members of PID structure and .DATA member of MSG structure.
bb *(optional)*	Bit number: 0 – 15 decimal Only applies to analog word members.

Example: MG59:33.DATA[0]/15

Note: For ControlNet device tags, the Tag Database editor supports the full addressing syntax and range supported for the PLC-5C programmable controllers.

SLC Addressing Syntax

The following section provides addressing syntax for the SLC 5 (Enhanced) programmable controllers, which include the SLC 5/03 (OS301) and SLC 5/04.

SLC 5 (Enhanced) I/O Files

F:ss.www/bb

F	File type: I = Input O = Output
ss	I/O slot number: 0 – 30 decimal
www *(optional)*	I/O word number expansion: 0 – 255 decimal
bb *(optional)*	Bit offset within word: 0 – 15 decimal When input slot is 0: 0 – 23 decimal

Example: I:22.254/13

Important: Writing to the output files of a SLC 5 is not recommended. However, if you do, be sure the SLC is not in RUN mode. If it is in RUN mode, the write will fail and an error message will be logged to the activity log file.

SLC 5 (Enhanced) Status Files

F:ww/bb

F	File type: S = Status
ww	Word address: SLC 5/03: 0 – 68 decimal SLC 5/04: 0 – 96 decimal
bb *(optional)*	Bit offset within word: 0 – 15 decimal

Example: S:15/6

SLC 5 (Enhanced) Binary, Integer, Float, ASCII, and String Files

Fnnn:www/bb

F	File type: B = Binary N = Integer F = Floating Point A = ASCII ST = String
nnn *(optional)*	File number: Binary: 3, 9 – 255 decimal Integer: 7, 9 – 255 decimal Floating Point: 8, 9 – 255 decimal ASCII: 9 – 255 decimal String: 9 – 255 decimal Binary, integer, and floating point file types use the default file number if the file number is absent. The default numbers are 3 (binary), 7 (integer), and 8 (floating point).
www	Word address: 0 – 255 decimal
bb *(optional)*	Bit offset within word: 0 – 15 decimal Bit offset is not supported for floating point and string file types.

Example: F8:129/2

SLC 5 (Enhanced) Binary Files: Optional Syntax

Fnnn/bbbb

F	File type B = Binary
nnn *(optional)*	File number: 3, 9 – 255 decimal Uses default file number 3 if the file number is absent.
bbbb	Bit offset from start of file: 0 – 4095 decimal

Example: B3/3999

SLC 5 (Enhanced) Timer, Counter, and Control Files

Fnnn:eee.MNE/bb

F	File type: T = Timer C = Counter R = Control
nnn *(optional)*	File number: Timer: 4, 9 – 255 decimal Counter: 5, 9 – 255 decimal Control: 6, 9 – 255 decimal Timer, counter, and control file types use the default file number if the file number is absent. The default numbers are 4 (timer), 5 (counter), and 6 (control).
eee	Element number: 0 – 255 decimal
MNE	Member mnemonic (see mnemonic tables on Page C–11)
bb *(optional)*	Bit number: 0 – 15 decimal Applies to analog members only.

Example: R67:123.EN

SLC 5 (Enhanced) Timer, Counter, and Control Files Bit Member Addressing: Optional Syntax

Fnnn:eee/MNE

F	File type: T = Timer C = Counter R = Control
nnn *(optional)*	File number: Timer: 4, 9 – 255 decimal Counter: 5, 9 – 255 decimal Control: 6, 9 – 255 decimal Timer, counter, and control file types use the default file number if the file number is absent. The default numbers are 4 (timer), 5 (counter), and 6 (control).
eee	Element number: 0 – 255 decimal
MNE	Bit member mnemonic (see mnemonic tables on Page C–11)

Example: C77:99/OV

SLC 5 (Enhanced) Timer, Counter, and Control Files Bit Member Addressing by Bit Address: Optional Syntax

Fnnn:eee/bb

F	File type: T = Timer C = Counter R = Control
nnn *(optional)*	File number: Timer: 4, 9 – 255 decimal Counter: 5, 9 – 255 decimal Control: 6, 9 – 255 decimal Timer, counter, and control file types use the default file number if the file number is absent. The default numbers are 4 (timer), 5 (counter), and 6 (control).
eee	Element number: 0 – 255 decimal

bb *(optional)*	Bit number: Timer: 13 – 15 Counter: 10 – 15 Control: 8 – 11, 13, 15

Example: T87:133/14

SLC 5 (Enhanced) Timer, Counter, and Control Files Analog Member Addressing: Optional Syntax

Fnnn:eee.o/bb

F	File type: T = Timer C = Counter R = Control
nnn *(optional)*	File number: Timer: 4, 9 – 255 decimal Counter: 5, 9 – 255 decimal Control: 6, 9 – 255 decimal Timer, counter, and control file types use the default file number if the file number is absent. The default numbers are 4 (timer), 5 (counter), and 6 (control).
eee	Element number: 0 – 255 decimal
o	Word offset: 1 – 2 decimal
bb *(optional)*	Bit number: 0 – 15 decimal

Example: R44:72.1/14

Mnemonic Tables

Timer Mnemonics

Mnemonic	Instruction	Type
ACC	Accumulator Value	Analog
DN	Done	Digital
EN	Enable	Digital
PRE	Preset Value	Analog
TT	Timing	Digital

Counter Mnemonics

Mnemonic	Instruction	Type
ACC	Accumulated Value	Analog
CD	Count Down Enable	Digital
CU	Count Up Enable	Digital
DN	Done	Digital
OV	Overflow	Digital
PRE	Preset Value	Analog
UN	Underflow	Digital
For fixed I/O SLC only:		
UA	Update Accumulator Value	Digital

Control Mnemonics

Mnemonic	Instruction	Type
DN	Done	Digital
EN	Enable	Digital
ER	Error	Digital
FD	Found	Digital
IN	Inhibit	Digital
LEN	Length	Analog
POS	Position	Analog
UL	Unload	Digital
For all programmable controllers except the fixed I/O SLC and the SLC 5/01:		
EM	Empty	Digital
EU	Enable Unloading	Digital

PID Mnemonics

Mnemonic	Instruction	Type	Submember Range
ADDR[]	Address of Master Loop %	Analog	0 – 3
BIAS	Output Bias %	Analog	
CA	Control Action	Digital	
CL	Cascaded Loop	Digital	
CT	Cascaded Type	Digital	
DATA[]	Reserved / Internal Use	Analog	0 – 13
DB	Deadband	Analog	
DO	Derivative Of	Digital	
DVDB	Deviation Alarm Deadband	Analog	
DVN	Deviation Alarm –	Analog	
DVNA	Deviation High Alarm	Digital	
DVP	Deviation Alarm +	Analog	
DVPA	Deviation Low Alarm	Digital	
EN	Enable	Digital	
ERR	Error	Analog	
EWD	Error Within Deadband	Digital	
INI	PID Initialized	Digital	
KD	Derivative Time	Analog	
KI	Integral Gain	Analog	
KP	Proportional Gain	Analog	
MAXI	Input Range Maximum	Analog	
MAXO	Output Limit High %	Analog	
MAXS	Setpoint Maximum	Analog	
MINI	Input Range Minimum	Analog	
MINO	Output Limit Low %	Analog	
MINS	Setpoint Minimum	Analog	
MO	Mode	Digital	
OLH	Output Limit High	Digital	
OLL	Output Limit Low	Digital	
OUT	Output	Analog	
PE	PID Equation	Digital	
PV	Process Variable	Analog	
PVDB	PV Alarm Deadband	Analog	
PVH	PV Alarm High	Analog	
PVHA	PV High Alarm	Digital	
PVL	PV Alarm Low	Analog	
PVLA	PV Low Alarm	Digital	
PVT	PV Tracking	Digital	
SO	Set Output %	Analog	
SP	Setpoint	Analog	

Mnemonic	Instruction	Type	Submember Range
SPOR	SP Out of Range	Digital	
SWM	Software A/M Mode	Digital	
TIE	Tieback %	Analog	
UPD	Update Time	Analog	

Message Mnemonics

Mnemonic	Instruction	Type	Submember Range
AD	Asynchronous Done	Digital	
AE	Asynchronous Error	Digital	
CO	Continuous	Digital	
DATA[]	Reserved / Internal Use	Analog	0 – 51
DLEN	Done Length	Analog	
DN	Synchronization Done	Digital	
EN	Enable	Digital	
ER	Synchronization Error	Digital	
ERR	Error Code	Analog	
EW	Enable Waiting	Digital	
NR	No Response	Digital	
RLEN	Request Length	Analog	
ST	Start Transmission	Digital	
TO	Time Out	Digital	

Block Transfer Mnemonics

Mnemonic	Instruction	Type
AD	Asynchronous Done	Digital
AE	Asynchronous Error	Digital
CO	Continuous	Digital
DLEN	Done Length	Analog
DN	Done	Digital
EC	Error Code	Analog
ELEM	Element Number	Analog
EN	Enable	Digital
ER	Error	Digital
EW	Enable Waiting	Digital
FILE	File Number	Analog
IDX	File Index	Analog
NR	No Response	Digital
PLEN	Program Length	Analog

Block Transfer Mnemonics (continued)

Mnemonic	Instruction	Type
RGS	Rack Group Slot	Analog
RLEN	Requested Length	Analog
RW	Read Writes	Digital
ST	Start	Digital
TO	Time Out	Digital
TOUT	Time Out	Analog

Token Data Mnemonics

Mnemonic	Instruction	Type
LO	Low	Analog
HI	High	Analog

SFC Status Mnemonics

Mnemonic	Instruction	Type
DN	Done	Digital
ER	Step Error	Digital
FS	Forced Scan	Digital
LS	Last Scan	Digital
OV	Timer Overflow	Digital
PRE	Preset	Analog
SA	Scan Active	Digital
TIM	Active Time	Analog

Remote I/O

The following section provides addressing syntax for Discrete and Block Transfer Remote I/O.

Discrete PLC Address

F:rrw/bb

F	File type: I = Input O = Output
rr	Rack number: PLC-2/20: 1 – 7 octal PLC-2/30: 1 – 7 octal PLC-3: 0 – 76 octal PLC-3/10: 0 – 17 octal PLC-5/11: 3 octal PLC-5/15: 1 – 3 octal PLC-5/20: 1 – 3 octal PLC-5/25: 1 – 7 octal PLC-5/30: 1 – 7 octal PLC-5/40: 1 – 17 octal PLC-5/60: 1 – 27 octal PLC-5/80: 1 – 27 octal PLC-5/250: 0 – 37 octal SLC5: 0 – 3 octal 1771-SB: 1 – 7 octal
w	Start word: 0 – 7 octal
bb *(optional)*	Start bit: 0 – 17 octal If none is specified, 0 is assumed.

Note: The address can also be specified as a length or a range. For length, use the format:

F:rrw/bb,ll
where ll is the bit length.
For range, use the format: F:rrw/bb – w/bb.

Examples:
I:070 is an input address pointing to word 0 on rack 07.
I:61/03 is an input address pointing to rack 6, word 1, bit 3.
O:011/01 is an output address pointing to bit 1 of word 1 in rack 1.
I:010/0,14 is an input address with a length specified.
I:010/0 – 0/17 is an input address with a range specified.

Block Transfer Address

Fff ww/bb

F	File type: B = Block
ff	File number: 1 – 64 decimal The file number *must* be followed by a space to separate it from the word number.
ww	Start word: 0 – 63 decimal **Note:** The actual PLC data table addresses will be determined by the block transfer instruction in the PLC program.
bb (*optional*)	Start bit: 0 – 15 decimal If none is specified, 0 is assumed.

Note: The address can also be specified as a length or a range. For length, use the format:

Fff ww/bb,ll
where ll is the bit length.
For range, use the format: Fff ww/bb – ww/bb

Examples:
B8 19 is file 8, word 19, bit 0.
Whether this is read or write depends on the definition of your block transfer instruction. Whether this is listen-only or full access depends on the definition for the rack containing the control byte.
B1 31/2,14 is a block address with a length specified.
B1 31/2 – 31/15 is a block address with a range specified.

Discrete SLC Address

F:S.ww/bb

F	File type: I = Input O = Output
S *(optional)*	Slot number: 0 – 7 decimal Not used by the PanelView terminal.
ww	Start word: 0 – 31 decimal
bb *(optional)*	Start bit: 0 – 15 decimal If none is specified, 0 is assumed.

Note: The address can also be specified as a length or a range. For length, use the format:

F:S.ww/bb,ll
where ll is the bit length.
For range, use the format: F:S.ww/bb – ww/bb

Examples:
I:.07 is an input address pointing to word 7.
O:2.01/01 is an output address pointing to bit 1 of word 1 on slot 2
I:.10/0,13 is an input address with a length specified.
O:.28/0 – 28/15 is an output address with a range specified.

Troubleshooting

This appendix contains information about problems you could encounter using the PanelBuilder software. It also contains causes of, and suggested responses to, PanelBuilder Error and Validation Error Messages you could encounter.

For information about troubleshooting file transfers, see the *PanelView 1200/1400e Transfer Utility User Manual*.

PanelBuilder Problems

The following table identifies some of the problems you could encounter using PanelBuilder software.

Problem	Cause	What to Do
Program won't run.	SHARE.EXE has not been installed.	For Windows 3.1, add the following line to your AUTOEXEC.BAT file: :share.exe/F:5100 /L:600
	A copy of PanelBuilder is already running.	Use the instance of PanelBuilder that is already running.
	Not enough Low Memory below 1 MB.	Close all other programs. You may need to run the MS-DOS memmaker utility to free up memory below 1 MB.
	Program crashed or terminated abnormally previously.	Restart Windows.
	Out of environment space.	Add the following line to your AUTOEXEC.BAT file: shell/e:4096
Unable to open shared database.	Database cannot be located.	Use "Repair Shared..." command to repair the link between the data file and application.
	Database is damaged, missing, or cannot be repaired.	Use "New Database..." command to remove the bad database. Then use "Import/Export..." or "Select Shared..." to re-populate the database tags.
	Current application is already using the database.	Choose a different database.
	Out of memory.	Close all Windows applications or other PanelBuilder applications to free up memory.
	8 databases are already open.	Close one or more PanelBuilder applications.
Unable to Open a New Application or Tag Database.	Not enough disk space.	Delete all unnecessary files on the hard disk. Exit Windows and delete any temporary files in the temporary directory.

Problem	Cause	What to Do
	Virtual Memory Swap file not large enough.	Windows 3.1: Use Control Panel, 386 Enhanced Settings to reconfigure the size and type of the swap file. We recommend using Permanent if 32-bit access is supported by your hardware. Windows 95: Choose the Performance tab from the System folder in the Control Panel and set the minimum size of the virtual memory swap file to 5 MB, and the maximum size to 16 MB. Windows NT: Choose the Performance tab from the System folder in the Control Panel. Click Change and set the Virtual Memory swap file to the Recommended number under Total Paging File Size for All Drives.
Tag Usage Viewer reports the wrong number of tags in the database.	Validation of an application stops after the first 100 errors. The Tag Usage Viewer reports the number of tags covered by the validation.	Correct the errors, then re-run the validation for accurate results.
Program is slow.	The computer is low on memory.	Close all other programs. Ensure that the swap file is configured to be the optimal size for your computer.
	The computer is low on resources.	Close all other programs. If this doesn't work, close down Windows and restart.
	Some other program is taking up a lot of processor time.	Find the offending program and shut it down.
Unable to add or edit tags in a database.	Database is damaged or made with a pre-release of the PanelBuilder software.	Use "Import/Export..." and save the existing tags to a PanelBuilder CSV format, then use "New Database..." to remove the old database and finally, use "Import/Export..." to recreate the existing tags.
	Cannot add tags.	Database already has a total of 32,000 Device and Remote I/O tags.
Application file won't download or upload.		For information about upload/download problems, see the *PanelView 1200e/1400e Transfer Utility User Manual.*
PanelView 1200/1400e Transfer Utility is performing another file transfer.	You are trying to upload/download a file while another upload/download is already in progress.	Close the Upload/Download dialog from the Transfer Utility if the transfer is complete.
PanelView 1200/1400e Transfer Utility is not installed. Please install PanelView 1200/1400e Transfer Utility Version 3 or Version 4 or later.	You are trying to upload/download a file, and the PanelView 1200/1400e Transfer Utility is not installed.	Install the PanelView 1200/1400e Transfer Utility Version 3 or Version 4 or later.
Installed Transfer Utility is not compatible with this version of PanelBuilder.	The version of the Transfer Utility installed on your computer is obsolete.	Install the PanelView 1200/1400e Transfer Utility Version 3 or Version 4 or later.
Tag Database won't start up.	Low resources.	Close all other programs and any extra PanelBuilder windows. If this doesn't work, close down Windows and restart.

Problem	Cause	What to Do
	Not enough disk space.	Delete all unnecessary files in the temporary directory. This may require running the MS-DOS deltree utility from within your temporary directory. You should close down Windows before doing this and be aware of what other items are in the temporary directory.
	Virtual Memory Swap file not large enough.	Windows 3.1: Use Control Panel, 386 Enhanced Settings to reconfigure the size and type of the swap file. We recommend using Permanent if 32-bit access is supported by your hardware. Windows 95: Choose the Performance tab from the System folder in the Control Panel and set the minimum size of the virtual memory swap file to 5 MB, and the maximum size to 16 MB. Windows NT: Choose the Performance tab from the System folder in the Control Panel. Click Change and set the Virtual Memory swap file to the Recommended number under Total Paging File Size for All Drives.

PanelBuilder Validation Errors

Consult the following table to identify and respond to PanelBuilder Validation Errors.

Validation Error Message	Cause	What to Do
Tag type must be Analog, Digital, or Block.	Incorrect tag type.	Use the correct tag type.
Tag [name] must be Analog or Digital.	Incorrect tag type.	Use the correct tag type.
Baud rate of [rate] is not valid for PLC device type [device type].	Some PLC types can only have a baud rate of 57.6K.	Consult your PLC User Documentation.
Object [object name] at [position on screen] overlaps a button.	Static element is on top of an input object (i.e., an indicator is placed over a push button).	Move one of these objects so they do not overlap.
Tag [tag name] does not exist.	You have entered an invalid tag name or the tag may have been deleted.	Check the tag definition in the database or add the tag to the database.
Tag [tag name] is not a device tag.	In the tag database, the device source is incorrect for the network type.	Check the tag definition.
Bad value in State List.	A state value is undefined.	Ensure that all the state values for the object are assigned and unique.
Tag [tag name] start value of [value] conflicts with another object's start value.	The object's initial state value for write tag conflicts with another object that uses the same write tag (but with a different initial state value).	Check the initial state for both objects to ensure they are both the same.
[Bitmap name] at [position on screen]: This image does not fit in the object for which it is being used as a label.	All labels must fit within their containing object.	Move or resize the object or the bitmap.
[Bitmap name] at [position on screen]: The image for this object is not present.	The image has been removed from the application or was never imported.	Check that the specified bitmap has been imported correctly.
[Object name] at [position on screen]: Part of the object is off–screen.	An object (or some element) is partly off-screen.	Reposition the object so it is completely on-screen.
[Symbol name] at [position on screen]: The image for this symbol must be monochrome.	A color bitmap is being used.	Import a monochrome bitmap.

Validation Error Message	Cause	What to Do
Number of visible states in scrolling object and cursor list must match.	The number of visible states does not match.	Change the number of visible states to match.
Object at [position on screen] has unassigned state values.	A state value is undefined.	Ensure that all state values for the object are assigned and unique.
A cursor list must exist if scrolling objects are used on the screen.	There is no cursor list on the screen.	Create a cursor list object or remove the scrolling object.
A cursor list and all the scrolling objects it controls must all be grouped together apart from other objects.	Cursor list object and its object lists must be in a single group.	Select the cursor list and all scrolling objects it controls and group them.
Cursor list can only be grouped with scrolling objects.	A non-scrolling object is grouped with the cursor list.	Remove any non-scrolling list objects from this group.
The group containing the cursor list cannot be embedded within another group.	The group with the cursor list and object lists is grouped with other objects.	Ungroup the scrolling objects from other objects before grouping them together.
Only one ASCII input object is permitted per screen.	More than one ASCII input object is on the screen.	Remove all but one ASCII input objects.
Only one cursor list object is permitted per screen.	More than one cursor list object is on the screen.	Remove all but one cursor list objects.
The text is longer than the maximum allowed.	The user-specified LARGE or double-wide font size of the Information Window does not permit a message that is greater than 35 characters long.	Reduce the font size using the Information Window Configurator to allow messages up to 70 characters long.
Tag entry not found.	The tag is not in the tag database.	Add the tag to the tag database or use another tag.
Not a device tag.	There is a problem with the tag.	Check the tag data source definition.
Not a RIO tag.	There is a problem with the tag.	Check the tag data source definition.
Cannot create tag.	There is a problem with the tag.	Check the tag definition.
Tag [tag name] does not exist.	The tag is not in the tag database.	Add the tag to the tag database.
Error creating configuration.	There is a problem with the PLC Communication setup.	Close the application and re-open it.
Cannot create channel.	There is a problem with the PLC Communication setup.	Close the application and re-open it.
Cannot create scan class.	There is a problem with the PLC Communication setup.	Close the application and re-open it.
There are more than 5000 used tags.	You have reached the maximum of 5000 assigned tags. The PanelView terminal only supports a maximum of 5000 tags.	Review your tags and remove any that are not being used.
Initial Value, Offset, or Scale factor is bad. Ensure de-scaled tag start value fits within address.	The analog tag initial value is too large to fit into the specified address.	Check the tag values or change the address and/or data type to include more bits.
	The state value for the object's initial state is too large to fit into the specified address.	Refer to Chapter 6, *Defining Tags.*
	The de-scaled initial value (or starting value) for analog tags that have offset or scale factor is too large to fit into the specified address.	Refer to Chapter 6, *Defining Tags.*
Target Firmware Version Exceeded.	The object/configuration you have defined is not compatible with the Target Firmware Version.	Reconfigure the application for compatibility with the current Target Firmware Version, or raise the Target Firmware Version of the application. See Appendix E, *Version Control,* for more information.
The image for this object is corrupt. Please re-import it.	Corrupted image file.	Re-import the image file.

Validation Error Message	Cause	What to Do
The associated write tag must also be assigned if Optional Write Expressions are used.	There is no assigned write tag associated with the optional write expression used.	Assign the write tag associated with the optional write expression for the object.
Only four Trend objects are allowed per screen.	You have assigned more than four Trend objects to the current screen.	Ensure that no more than four Trend objects are used on the current screen.
No more than 3000 points are allowed in all the Background plotting trend pens in an application.	The total number of data points for all pens using background plotting in your application cannot exceed 3000.	Configure fewer pens for background plotting, or reduce the sample rate or time span of some of the trend objects.
No more than 85 pens are allowed in all the Background plotting trend objects in an application.	You can configure a maximum of 85 pens (across all Trends in your application) for background screen plotting.	Reduce the number of pens configured for background screen plotting in your application.

Version Control

What Is Version Control?

Version Control is a set of rules embedded in PanelBuilder and in PanelView terminal firmware, which determines the version of PanelView firmware with which a PanelBuilder application is compatible.

Version Control is imposed at three levels:

- when a PanelBuilder application is validated
- when a PanelBuilder application is downloaded to a PanelView terminal
- before a PanelView terminal application is made current or executed at a PanelView terminal

Version Control and Application Validation

Before a downloadable (.pvd) application file can be created, it must be validated by PanelBuilder. Part of this validation ensures that the features used in the application (the application's Actual Firmware Version) do not exceed the application's Target Firmware Version.

Target Firmware Version vs. Actual Firmware Version

To understand the difference between an application's Target Firmware Version and its Actual Firmware Version, consider the following examples:

- A PanelBuilder ControlNet application's Target Firmware Version is set to 2 and later. If the application does not make use of features which require Version 2 firmware, the application's Actual Firmware Version is 1.03, and it will run on terminals with firmware Version 1.03 (required for ControlNet communications) and later. The application will not run on terminals with Version 1 firmware, however.
- A PanelBuilder ControlNet application's Target Firmware Version is set to 2 and later. If the application does make use of features which require Version 2 firmware, its Actual Firmware Version is 2 and it will not run on terminals equipped with firmware versions lower than 2.
- A PanelBuilder DH+ application's Target Firmware Version is set to 1 and later. If the application makes use of features which require Version 2 firmware, its Actual Firmware Version is 2. Validation fails, because the features used in the application exceed the application's Target Firmware Version.

The Target Firmware Version is therefore an upper threshold for the features used in an application, and causes a validation error if that threshold is exceeded. This helps ensure that the applications you develop with PanelBuilder will run on the terminals to which they are downloaded.

Version 1.03 Firmware Features

PanelView Version 1.03 firmware includes support for all the features of Version 1 firmware, plus:

- ControlNet communications
- Using ControlNet for a Pass-Through download

Version 2 Firmware Features

PanelView Version 2 firmware includes support for the following PanelBuilder features:

- Expressions
- Local Messages longer than 72 characters
- Alarm Messages longer than 50 characters
- Information Messages longer than 35 characters, with a double-wide or large font
- Information Messages longer than 70 characters, with a small or double-high font
- Embedded Newline control characters (\n) in Local Messages
- Embedded Newline control characters (\n) in Information Messages
- Embedded Newline control characters (\n) in Alarm Messages
- Remote I/O Floating Point analog tags
- Numeric Input Objects configured with the Implicit Decimal Option
- Extra large fonts
- Skipped states used in Scrolling Lists
- Use of the Retain Last List State feature in Scrolling Lists
- Cursor List Preview States of the value
  0 to (((*Number of Visible States* + 1) / 2) – 1).
  In terminal firmware versions earlier than version 2, the range was
  0 to (*Number of Visible States* / 2) – 1.
- SLC slot number configured to a value other than 0
- The PLC Controlled Alarm Acknowledge Control
- Multiple Alarm Controls
- The two Alarm Message to PLC Controls: the Message Handshake Expression and the Message Control Tag
- Device block tags
- Trend objects

- Scale objects
- Alarm window configured as a panel
- Increment/Decrement buttons with floating point increments/decrements

Version 3 Firmware Features

PanelView Version 3 firmware includes support for the following PanelBuilder features:

- Goto Configure Mode button
- PanelView 1000e applications
- Retain Cursor On Cancel feature
- Modbus Communications (with the Modbus Communications Kit, catalog number 2711E-UMOD)

Version 4 Firmware Features

PanelView Version 4 firmware includes support for the following features:

- Ethernet Pass-Through over Remote I/O
- 1500 Message Alarm History

Version Control and Downloading Applications

Because you can use PanelView terminals as PCMCIA card writers for other terminals, you can download applications to terminals which cannot run them.

Version Control does ensure, however, that you do not attempt to Make Current any application which will not run on the terminal to which you are downloading.

If you have selected the Make Current Application option in the Setup Download dialog box before downloading an application to a PanelView terminal, PanelBuilder terminates the download with an error under these circumstances:

- The firmware version required by the features used in the application (the Actual Firmware Version) exceeds the firmware version installed in the PanelView terminal to which you are downloading.
- The terminal is of the wrong type. The download aborts with an error if:

The application is configured for this:	And the terminal to which you're downloading is this:
any Touch Screen terminal	any Keypad terminal
any Keypad terminal	any Touch Screen terminal
1400e Touch Screen terminal	1200e Touch Screen terminal, any Keypad terminal
1400e Keypad terminal	1200e Keypad terminal, any Touch Screen terminal

- A 1784-KTCX network communication card is not installed in a terminal to which you're downloading a ControlNet application.

If you have not selected the Make Current Application option in the Setup Download dialog box before downloading an application to a PanelView terminal, PanelBuilder downloads the application, but warns you if any of the three conditions mentioned above exist.

Version Control at the PanelView Terminal

Because you can use PanelView terminals as PCMCIA card writers for other terminals, you can download applications to terminals which cannot run them.

Version Control does ensure, however, that you do not attempt to Make Current any application which will not run on the terminal to which you are downloading.

The PanelView terminal will not allow you to select the Make Current Application option for an application under these circumstances:

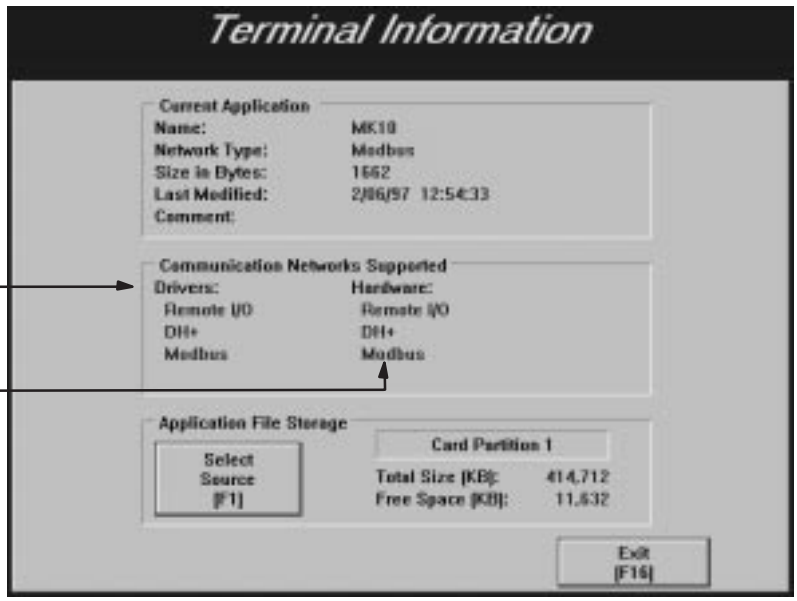
- The firmware version required by the features used in the application (the Actual Firmware Version) exceeds the firmware version installed in the PanelView terminal.
- The terminal is of the wrong type. You can select the Make Current Application option for these configurations:

The application is configured for this:	And the terminal is this:
1000e Touch Screen terminal 1200e Touch Screen terminal 1400e Touch Screen terminal	1000e Touch Screen terminal 1400e Touch Screen terminal
1200e Touch Screen terminal	1200e Touch Screen terminal
1000e Keypad terminal 1200e Keypad terminal 1400e Keypad terminal	1000e Keypad terminal 1200e Keypad terminal 1400e Keypad terminal

- The required communication hardware or firmware driver has not been installed in the PanelView terminal. For example, for ControlNet, a 1784-KTCX network communications card *and* the correct communication driver must be installed in the terminal. For more information about determining the correct configuration of the terminal, see the *PanelView 1000e, 1200e, and 1400e Operator Terminals User Manual.*
- The correct network communication driver is not installed. You can verify that the correct hardware and communication drivers are installed by referring to the Terminal Information screen when the terminal is in Configure Mode.

This list indicates the firmware drivers that are installed.

This list indicates the hardware devices that are installed and detected.



Reserved Keywords

The following keywords are reserved for use in the Expression language, and cannot be used to name local variables:

And	Else	Integer	SelectCase
As	End	Long	Single
Boolean	EndIf	Mod	Then
Byte	EndSelect	Not	To
Case	Exit	Or	True
CaseElse	False	Rem	Variant
Dim	If	Select	Xor

The following keywords are reserved for future use in the Expression language, and cannot be used to name local variables:

ByRef	Function	Static	While
ByVal	Loop	Step	
Do	LoopUntil	String	
Double	LoopWhile	Sub	
DoUntil	Next	Ulong	
DoWhile	Private	Until	
For	Public	Wend	

Symbols

.amg file, 10–11
.apl file, 3–4
.cfg file, 3–4
.csv file, 6–33, 6–35
  and column ordering, 6–37
  importing a database, 6–36
.db file, 6–33
.img file, 10–11
.lmg file, 10–11
.pbw file, 3–4
.pvc file, 3–4
.pvd file, 3–4
? (user entry character), 7–6
  in write expressions, 7–24

Numbers

1771 Remote I/O Link, 1–5, 5–11
1785-KA, 2–39
  bridging DH+ networks, 2–39, 5–6, 5–19
6200 Series Programming Software, 2–40

A

A.I. Series software, importing a database, 6–36
Acknowledge button hold time, 12–7
Acknowledge hold time, 12–8
Acknowledge to PLC controller tag, 14–5
Acknowledging alarms, 11–5
Actual firmware version, defined, E–1
Addresses
  See also Offlink address
  and tags, 2–13, 2–16, 2–19, 2–43
  assigning, 2–43, 5–21
  block transfer file, 2–16, 5–10
  discrete I/O, 2–16
  planning, 2–19
    using worksheets, 2–19
  recording on worksheets, 2–27, 2–50
  reducing requirements, 2–18
  retentive objects, 9–4
  station, 2–37
    local, 2–38
    remote, 2–38
Addressing syntax
  PLC-2, C–1
  PLC-3
    data table sections, C–2
    timers and counters, C–2
  PLC-5
    binary files: optional syntax, C–3
    data files, C–3
    I/O and status files, C–4
    timers, counters and control files, C–4
  PLC-5 (Enhanced)
    binary files: optional syntax, C–6
    binary, integer, BCD, ASCII, float and string files, C–5
    I/O and status files, C–5
    timer, counter, control, SFC status, message, PID, block transfer and token data files, C–6
  SLC 5, writing to output files, C–7
  SLC 5 (Enhanced)
    binary files: optional syntax, C–9
    binary, integer, float, ASCII and string files, C–8
    I/O files, C–7
    status files, C–8
    timer, counter and control files, C–9
      analog member addressing: optional syntax, C–11
      bit member addressing: optional syntax, C–10
Alarm acknowledge to PLC, about, 11–8
Alarm control delay, 12–7
Alarm controls, 11–9
Alarm editors, 11–5
Alarm history screen, 11–50
  about, 11–3, 11–50
  calling up from the terminal, 11–30, 11–34, 11–50
  configuring, 11–52
  screen security, 8–8
  using buttons, 11–51
  using in the terminal, 11–50
Alarm message panel, about, 11–33
Alarm message window
  about, 11–3, 11–29
  choosing type
    multi-line, 11–36
    single-line, 11–36
  disabling, 11–36
  enabling, 11–36
  multi-line, 11–31

single-line, 11–30
types, 11–29
using buttons, 11–32
using in the terminal, 11–30

Alarm messages, 10–1, 11–29, 11–31
audio, 11–23
changing options, 10–9
displaying on the terminal, 11–31
enabling
audio option, 10–8
print option, 10–8
the alarm relay, 10–8
in alarm history screen, 11–50
in alarm status screen, 11–41
in alarm window, 10–8
options, 10–8

Alarm status screen, 11–39
about, 11–4, 11–39
assigning reset control
automatic, 11–49
manual, 11–49
calling up from the terminal, 11–41
configuring, 11–48
display active alarms mode, 11–42
display all messages mode, 11–43
display past alarms mode, 11–43
resetting, 11–46
screen security, 8–8
using buttons, 11–45
using in the terminal, 11–40
viewing alarm messages, 11–41
viewing modes, 11–41

Alarm triggers
about, 11–9
bit, 11–6
word, 11–6

Alarms
about, 11–1
acknowledging, 11–5
background, 11–29
configuring, 11–5
displaying, 11–3
global controls, configuring, 11–6
message control, 11–28
message handshake, 11–28
message to PLC, 11–24
methods of indicating, 11–1
PLC controlled audio, 11–22
PLC controlled relay, 11–23
PLC controlled silence alarms, 11–22
quantity and time, 11–44
silence alarms to PLC, 11–22
triggering, 11–2

Aligning, objects, 9–43

Allen-Bradley
1771 Remote I/O Link, 1–5
6200 Series Programming Software, 2–40
technical support, P–10

Analog tags, 6–2
address length, 6–23
configuring, 6–13
data types, 2–24, 2–47

AND, in expressions, 7–17

Application file comment, 12–3

Application startup screen
assigning, 8–15
changing, 8–15
security, 8–8
selecting, 12–2

Applications
building, 2–1
closing, 3–17
compressing, 3–8, 3–9
configuring, 2–1
converting
from PanelBuilder 1200, 3–4
from PanelBuilder for DOS, 3–4
copying, 3–10
creating, 2–1, 3–1
defining communications, 5–1
ControlNet, 5–15
DH+, 5–15
Remote I/O, 5–2
deleting, 3–17
describing, 12–3
determining size, 3–11
downloading, 2–1, 15–1
file formats, 3–4
opening, 3–3
optimizing, 2–4
planning, 2–1, 2–2
ControlNet, 2–5
DH+, 2–5
Remote I/O, 2–5
renaming, 3–10
running, 1–2
saving, 3–6, 3–7
in another file format, 3–8
testing, 2–1
transferring, 15–1
from within PanelBuilder, 15–1
using the PanelView 1200/1400e Transfer Utility, 15–3
uploading, 15–1
validating, 3–10

Applying filters, in Tag Usage Viewer, 6–26

Arcs, 9–5, 9–12, 9–21, 9–33

Arithmetic operations, in expressions, 7–9

Arranging, inner text, 9–46
ASCII display, 9–5, 9–7, 9–21
ASCII input, 9–5, 9–7, 9–21
Assigning
  device type, 5–8
  racks, 5–8, 5–9
  write expressions, 7–25
Auto-repeat functions
  rate, 12–8
  start delay, 12–8

B

Background alarms, 11–29
Background color, screens, 8–7
Bar graph, 9–5, 9–7, 9–21
Baud rate
  for remote I/O, 5–3
  selecting on worksheets, 2–32
BCD data type, 2–25, 2–49
Binary Coded Decimal data type. *See* BCD data type
Binary data type, 2–24, 2–48
Bit position data type, 2–25, 2–48
Bit-triggered alarms, 11–6
  about, 11–7
Bitmap image, 9–21
Blinking objects, 9–49
Block tags, 6–2
  configuring, 6–18
Block Transfer File I/O Usage Worksheet, A–1
  using, 2–29
Block Transfer File Worksheet, A–1
  using, 2–33
Block transfer files
  about, 5–10
  access, 5–12
  addresses, 2–16
  changing file assignments, 5–14
  creating, 5–12
  deleting file assignments, 5–14
  file number, 5–12
  file size, 5–12
  requirements, 2–18
  setting up, 5–10
Block transfer mnemonics, C–14, C–15
Borders, 9–48
Boxes, 9–6, 9–21
Bridged DH+ networks
  remote station address, 5–6
  with 1785-KA, 2–39, 5–6, 5–18
  with Pyramid Integrator, 2–40, 5–6, 5–19
Buttons
  alarm history screen, 11–51
    acknowl alarm, 11–51
    alarm status, 11–51
    cursor up/cursor down, 11–51
    exit, 11–51
    page up/page down, 11–51
    print, 11–51
    silence alarms, 11–51
  alarm message window, 11–32
    acknowl alarm, 11–32
    alarm history, 11–32
    alarm status, 11–32
    cursor up/down, 11–32
    exit, 11–32
    silence alarms, 11–32
  alarm status screen, 11–45
    alarm history, 11–45
    cursor up/cursor down, 11–45
    display mode, 11–45
    exit, 11–45
    page up/page down, 11–45
    print, 11–45
    reset Qty/Time, 11–45
    silence alarms, 11–45
  auto-repeat functions, 12–8
Byte data type, 2–24, 2–48

C

Cable, upload/download, 1–6
Changing
  block transfer file assignments, 5–14
  network type, 5–25, 5–26, 5–30
  rack assignments, 5–14
Channel number, 5–22
Checking syntax, in expressions, 7–23
Clarity, and screen design, 2–6
Closing
  applications, 3–17
  screens, 8–7
Color palette, using, 9–13
Commands, conventions, P–9
Communication errors, time before reporting, 5–23
Communication Setup editor, 5–2, 5–15
Communications, defined, 5–1
Comparison operations, in expressions, 7–11

Compatibility, version, 1–2
Compilation, of expressions, 7–20
Components, in expressions, 7–5, 7–21
  maximum number of, 7–8
Compressing, applications, 3–8, 3–9
Configuring
  alarm history screen, 11–52
  alarm status screen, 11–48
  alarms
    configuring, 11–5
    global controls, 11–6
  analog tags, 6–13
  block tags, 6–18
  communications, 2–1
  digital tags, 6–16
  information message window, 14–1, 14–3
  list object states, 9–59
  nodes, 5–22
  object defaults, 12–9
  object states, 9–55, 9–56, 9–57
  objects, 9–53
  scan classes, 2–42, 5–24
  string tags, 6–17
  tag type, 6–13
  terminal setup, 12–1
Consistency
  and screen design, 2–6
  using screen templates, 2–7, 8–3
Construct operations, in expressions, 7–12
  IF..THEN..ELSE, 7–13
  SELECT CASE, 7–15
Control byte, 5–13
Control list selector with enter, 9–5, 9–7, 9–21
Control list selector without enter, 9–5, 9–7, 9–21
Control mnemonics, C–12
Control selectors, 9–7
  control list selector with enter, 9–7
  control list selector without enter, 9–7
  set bit cursor point, 9–7
ControlNet
  choosing device as data source, 6–3
  Node Editor, 5–15, 5–21
  optimizing data flow, 5–24
  Scan Class Editor, 5–15
  scan classes, 5–24
  setting up terminal communications, 5–16
  station number, 5–17
ControlNet Communications, A–1
ControlNet Communications Worksheet, using, 2–36
Controls
  alarms, 11–9
  copying with screen object, 8–5
Conventions
  commands, P–9
  documentation, P–7
  key combinations, P–9
  mouse, P–7
  print, P–7
  selection, P–7
  selection in dialog boxes, P–8
  shortcut keys, P–9
Converting
  PanelBuilder 1200 applications, 3–4
  PanelBuilder for DOS applications, 3–4
  private tag database to shared, 6–33
  tags, 5–26
Copying
  applications, 3–10
    with shared tag database, 3–10
  expressions, 7–23
  objects, 9–36
  screen contents, 8–4
  screens, 8–3
Counter mnemonics, C–12
Creating
  applications, 2–1, 3–1
  arcs, 9–33
  block transfer files, 5–12
  expressions, 7–1
  extended characters, 9–25
  folders, 6–7
  graphic image, 9–29
  labels, 9–34
  lines, 9–33
  messages, 10–3, 10–6
  objects, 9–20, 9–22
  panels, 9–33
  reports, 3–12
  symbols, 9–26
  screens, 8–1
  tag folder, 6–4
  tags, 6–1
  text, 9–24
  wedges, 9–33
Cursor, retain on cancel, 12–10
Cursor visibility, configuring retain cursor on cancel, 12–10
Cutting, objects, 9–36

D

Data flow, optimizing for ControlNet, 5–24

Data Highway Plus. *See* DH+

Data sources, 6–2, 6–20
- and relaxed validation, 6–21
- device, 6–2, 6–20, 6–21
  - and scan classes, 5–24
- Remote I/O, 6–23
- remote I/O, 6–2, 6–21
- specifying, 6–21, 6–23

Data types, 2–24, 2–47
- BCD, 2–25, 2–49
- binary, 2–24, 2–48
- bit position, 2–25, 2–48
- byte, 2–24, 2–48
- default, 2–24, 2–48
- floating point, 2–24, 2–48
- in expressions, 7–3
- long integer, 2–24, 2–48
- signed integer, 2–24, 2–48
- unsigned integer, 2–24, 2–48

Date and time, in PanelView terminal
- date and time to PLC controller data tags, 13–6
- PLC controlled date and time control tag, 13–4

Date and time to PLC controller data tags, 13–6

Date display, 9–5, 9–13, 9–21

Declaring, local variables, 7–6

Decrement value button, 9–5, 9–9, 9–21

Default data type, 2–24, 2–48

Defaults
- assigning, 9–52
- file, deleting, 9–53
- object defaults, 12–9

Defining
- communications, 5–1
  - for ControlNet applications, 5–15
  - for DH+ applications, 5–15
  - for Remote I/O applications, 5–2
- DH+/RIO Pass-Through control byte, 5–13
- Ethernet Pass–Through, 5–13
- network download controls, 5–20
- PanelView Terminal Address, 5–19
- PLC station address, 5–7

Deleting
- applications, 3–17
- block transfer file assignments, 5–14
- data in spreadsheets, 4–12
- folders, 6–9
- module group from a rack, 5–14
- objects, 9–37
- rack assignments, 5–14
- screens, 8–7

Describing applications, 12–3

Designing
- screen templates, 2–7
- screens, 2–6, 2–8

Determining application size, 3–11

Device
- choosing as data source, 6–21
- data source
  - for ControlNet applications, 6–3
  - for DH+ applications, 6–3
  - for Remote I/O applications, 6–3

Device data source, and scan classes, 5–24

Device Tags Worksheet, A–1
- recording object information, 2–45
- recording tag information, 2–46
- recording the address, 2–50
- using, 2–43

Device type
- assigning, 5–8
- specifying, 5–9

DH+
- and PanelView, 1–6
- choosing device as data source, 6–3
- Nodes editor, 5–21
- remote station address
  - bridged by 1785-KA, 2–39
  - bridged by Pyramid Integrator, 2–40
- scan classes, 5–24
- Scan Classes editor, 5–15
- setting up terminal communications, 5–16
- station number, 5–17

DH+ Communications Worksheet, A–1
- defining the nodes, 2–34, 2–35, 2–37, 2–50
- using, 2–36

DH+/RIO Pass-Through
- assigning, 2–34
- block transfer control byte, 5–13

Dialog box editors
- about, 4–1
- navigating, 4–1

Dialog boxes
- about, 1–1
- selection conventions, P–8

Digital tags, 6–2
- configuring, 6–16

Disabling, nodes, 5–23

Discrete I/O addresses, 2–16

Discrete I/O Usage Worksheet, A–1
using, 2–28
Display active alarms mode, 11–42
Display all alarms mode, 11–43
Display past alarms mode, 11–43
Displaying alarms, 11–3
Downloading applications, 2–2, 15–1
version control and, E–3
Drawing
arcs, 9–33
graphic images, 9–29
lines, 9–33
objects, 9–20
panels, 9–33
wedges, 9–33
Duplicating
folders, 6–9
objects, 9–37
rows in spreadsheets, 4–11
tags, 6–11
DXF Image, 9–21
Dynamic objects, 9–2
read, 9–2
write, 9–2

E

Editing
messages, 10–3
objects, 9–35
selections in spreadsheets, 4–12
tags, 6–11
text, 9–25
Editors
about, 1–1
alarms, 11–5
Communication Setup, 5–2, 5–15
ControlNet Nodes, 5–15, 5–21
ControlNet Scan Classes, 5–15
DH+ Nodes, 5–15, 5–21
DH+ Scan Classes, 5–15
dialog box, 4–1
form/spreadsheet, 4–3
RIO Block Transfer Files, 5–2, 5–11
RIO Racks, 5–2, 5–8
Scan Classes, 5–24–5–25
Tag Database, 6–5
types, 4–1
working in multiple, 6–12
Ellipse, 9–5, 9–21
Ellipses, 9–12
Enabling, nodes, 5–23
Enhancing, PanelView terminals, Series F and G, 1–2
Enter key parameters, 12–8
control delay, 12–8
handshake time, 12–8
Epson FX-80 compatible printers, 1–7
Equipment
optional, 1–6
required, 1–3
Errors
communication, time before reporting, 5–23
syntax, in expressions, 7–23
Ethernet Pass–Through
assigning, 2–34
block transfer control byte, 5–13
setting up, 5–5
Evaluation
of expressions, 7–19
of local variables, 7–7
Examples, expressions, 7–28
Exporting
graphic images, 9–31
messages, 10–10, 10–13
symbols, 9–28
tag database, 6–35
to MODSOFT database, 6–36
to Taylor ProWORXPLUS database, 6–36
Editors, Expression, 7–21
Expressions
about, 7–2
checking syntax, 7–23
compilation of, 7–20
components, 7–5, 7–21
maximum number of, 7–8
creating, 7–1
cutting, copying and pasting, 7–23
data types, 7–3
defined, 7–1
editing, 7–21
after compilation, 7–20
elements removed from during compilation, 7–20
errors, syntax, 7–23
examples, 7–28
floating point accuracy, 7–4
formatting, 7–22
how they work, 7–2
indenting, 7–23
language, about, 7–4
limits in, 7–8
local variables in, 7–6

maximum size of, 7–8, 7–20
operations, order of precedence, 7–4
operations in, 7–4
optional write, 7–2
read, 7–2
regeneration of, 7–20
reserved keywords, F–1
return values in, 7–19
samples, 7–28
syntax
checking, 7–23
errors, 7–23
types, 7–2
user entry character (”?”), 7–6
validating, 7–27
values, 7–19
write, 7–2, 7–24

Extended character set, B–1
MS Word Sans Serif, 9–25
using, 9–26
PanelView, 9–25
using, 9–25

F

Features, new, P–1

File format, for importing messages, 10–11

Filtering data
in the Tag Usage Viewer, 6–26
removing filter, 6–28
setting filters, 6–28

Firmware, target version, 12–4

Firmware features
Version 1.03, E–2
Version 2, E–2
Version 3, E–3
Version 4, E–3

Flipping, objects, 9–45

Floating point, accuracy in expressions, 7–4

Floating point data type, 2–24, 2–48

Folders
adding tags, 6–9
creating, 6–7
deleting, 6–9
digital tags, 6–17, 6–18
duplicating, 6–9
nesting, 6–4, 6–8
opening multiple folders, 6–8
using to group tags, 6–4
working with, 6–7

Font, in information message window, 14–4

Font sizes, 9–50

Form/Spreadsheet editors, using, 4–3

Formatting, expressions, 7–22
indenting, 7–23

Formatting codes, in messages, 10–4

Forms, using, 4–3

Fractional racks. *See* Racks, partial

Full access, 2–16

Full racks. *See* Racks, full

Function keys
assigning, 9–54
displaying in PanelBuilder, 8–14

G

Global controls
message control, 11–28
message handshake, 11–28
message to PLC, 11–24
PLC controlled audio, 11–22
PLC controlled relay, 11–23
PLC controlled silence alarm, 11–22
silence alarms to PLC, 11–22

Goto Configure Mode, 9–8

Goto screen button, 9–5, 9–11, 9–21
for calling up the Alarm History screen, 11–30, 11–34

Graphic images, 9–5, 9–8, 9–29
as labels, 9–34
exporting, 9–31
importing, 9–27, 9–30, 9–32
library, 9–29
removing, 9–28, 9–31

Graphic objects. *See* objects

Grids
changing default settings, 8–12
screen, 8–11
touch cell, 8–13
using, 8–11

Grouping
See also Outer text
nested groups, 9–41
objects, 9–41
performance of PanelBuilder, 9–41
tags, 6–4
ungrouping objects, 9–41

Groups, ungrouping nested groups, 9–41

H

Hardware, requirements, 1–3

Help. *See* Online help

Hot keys. *See* Shortcut keys

I

I/O
  image table, 2–13, 2–28
  racks, 2–13
IF..THEN..ELSE, in expressions, 7–13
Image
  bitmap, 9–21
  DXf, 9–21
Import, graphic images, 9–27, 9–30, 9–32
Importing
  messages, 10–9
    file extensions, 10–11
    guidelines, 10–10
    using clipboard, 10–9
  tag database, 6–35
    from A.I. Series database, 6–36
    from Logic 5 database, 6–36
    from MODSOFT database, 6–36
    from PanelBuilder 900 database, 6–36
    from Taylor ProWORXPLUS database, 6–36
Increment value button, 9–5, 9–9, 9–21
Indenting, in expressions, 7–23
Indicators, 9–8
Information message window
  about, 14–1
  acknowledging, 14–2
  clearing, 14–2
  configuring, 14–3
  configuring window removal, acknowledge hold time, 12–8
  disabling, 14–3
  enabling, 14–3
  font type, 14–4
  removing, 14–4
  tags, 14–5
  using at the terminal, 14–2
Information messages, 10–1
Information window control tag, 14–5
Initial state value, 2–22, 2–27, 2–45, 2–49, 9–3, 9–60
  deleting objects with, 8–7
Initial value, 2–27, 2–49, 9–3
Inner text, 9–24
  arranging, 9–46
Inserting, rows in spreadsheets, 4–12
Interlocked push button, 9–5, 9–10, 9–21

K

KA, modules, 2–40
Key conventions, P–9
Keypad objects, 9–1
Keywords, reserved, F–1

L

Labels, 9–29, 9–34
Last chassis, 2–16, 5–10
Last Quantity/Time Reset stamp. *See* Alarm status reset time/date stamp
Latched push button, 9–5, 9–10, 9–21
Limits, in expressions, 7–8
Lines, 9–5, 9–12, 9–21, 9–33
  changing styles, 9–48
List indicator, 9–5, 9–8, 9–21
List objects, 9–59
  resizing, 9–59
Listen Only access, 2–16
Local message display, 9–5, 9–9, 9–21
Local messages, 10–2
Local station address, 2–38
Local variables
  assigning values to, 7–7
  declaring, 7–6
  evaluation of, 7–7
  in expressions, 7–6
    naming, 7–6
  syntax, 7–7
  reserved keywords, F–1
Logic 5 software, importing a database, 6–36
Logical operations, in expressions, 7–16
Long integer data type, 2–24, 2–48

M

Maintained push button, 9–5, 9–10, 9–21
Maximums, components, number of in expressions, 7–8
Message control, 11–28
Message handshake, 11–28
Message to PLC, 11–24
Messages
  about, 10–1
  alarm, 10–1
  creating, 10–3, 10–6
  editing, 10–3

editors, 10–1
exporting, 10–10, 10–13
file format for importing, 10–11
formatting codes in, 10–4
importing, 10–9
file format for, 10–11
using clipboard, 10–9
information, 10–1
local, 10–2
mnemonics, C–14
saving, 10–14
trigger value, 10–2
updating, 10–14
validating, 10–13

Mnemonics
block transfer, C–14, C–15
control, C–12
counter, C–12
message, C–14
PID, C–13
SFC status, C–15
tables, C–12
timer, C–12
token data, C–15

Module groups, 2–14, 5–10
deleting from a rack, 5–14

Modules
KA, 2–40
RM, 2–40

Momentary normally
closed push button, 9–10, 9–21
open push button, 9–10, 9–21

Mouse conventions, P–7

Moving, objects, 9–39

Multi-component objects, 9–51

Multi-line alarm message window, 11–31
choosing, 11–36
using buttons, 11–32

Multistate indicator, 9–5, 9–9, 9–21

Multistate objects, 9–55

Multistate push button, 9–5, 9–10, 9–21

N

Naming, tags, 6–4

Nesting folders, 6–4, 6–8

Network download controls
transfer inhibit control, 5–20
transfer request control, 5–20
transfer status control, 5–20

Network file transfers
ControlNet
PanelView terminal address, 5–18
setting up, 5–18
DH+
PanelView terminal address, 5–18
setting up, 5–18
Remote I/O
PLC station address, 5–4, 5–5
setting up, 5–4

Network transfer controls
defining, 5–20
Transfer inhibit, 5–7
Transfer request, 5–7
Transfer status, 5–7

Network type, changing, 5–25, 5–26, 5–30
from ControlNet to DH+, 5–34–5–35
from ControlNet to Remote I/O, 5–33–5–35
from DH+ to ControlNet, 5–31–5–35
from Remote I/O to ControlNet, 5–28–5–35

Node, name, valid characters, 5–22

Nodes
about, 5–21
configuring, 5–22
defining, 2–34, 2–35, 2–37, 2–50
enabled, 5–23
local station addressing, 2–38
name, 2–37
PLC type, 5–22
remote station addressing, 2–38
station address, 5–22
timeout, 5–23

Non-retentive objects, 9–3

Normally closed momentary push button, 9–5

Normally open momentary push button, 9–5

NOT, in expressions, 7–17

Numeric data display, 9–5, 9–9, 9–21

Numeric entry keypad, 9–5, 9–9, 9–13

Numeric input cursor point, 9–5, 9–9, 9–22
retain cursor on cancel, 12–10

Numeric keypad-enable button, 9–6, 9–10, 9–22

Numerics, 9–9

O

Object setup, configuring, 12–9

Objects
addresses, 2–19
aligning, 9–43
arcs, 9–5, 9–12, 9–21, 9–33
arranging, 9–40
ASCII display, 9–5, 9–7, 9–21

ASCII input, 9–5, 9–7, 9–21
assigning addresses, 2–43
assigning function keys, 9–54
assigning tags, 9–61
assigning write expressions to, 7–24
bar graph, 9–5, 9–7, 9–21
blinking, 9–49
boxes, 9–6, 9–21
buttons, changing touch margins, 9–50
changing border styles, 9–48
changing colors, 9–47
changing line styles, 9–48
colors, 9–47
commands, 9–14, 9–21
configure dialog box, 9–53
configuring, 9–53
control list selector with enter, 9–5, 9–7, 9–21
control list selector without enter, 9–5, 9–7, 9–21
copying, 9–36
  with addresses, 9–36
  without addresses, 9–36
creating, 9–20, 9–22
cutting, 9–36
date display, 9–5, 9–13, 9–21
decrement value button, 9–5, 9–9, 9–21
defaults, 9–52
  deleting, 9–53
defined, 9–1
deleting, 9–37
drawing, 9–20
duplicating, 9–37
dynamic, 9–2
  read, 9–2
  write, 9–2
editing, 9–35
ellipse, 9–5, 9–21
ellipses, 9–12
fill, 9–48
flipping, 9–45
Goto Configure Mode, 9–8
goto screen button, 9–5, 9–11, 9–21
graphic images, 9–5, 9–8, 9–29
grouping, 9–40
increment value button, 9–5, 9–9, 9–21
initial state value, 2–27, 2–49, 8–7, 9–3, 9–60
interlocked push button, 9–5, 9–10, 9–21
keypad, 9–1
latched push button, 9–5, 9–10, 9–21
lines, 9–5, 9–12, 9–21, 9–33
list indicator, 9–5, 9–8, 9–21
list states, configuring, 9–59
local message display, 9–5, 9–9, 9–21
maintained push button, 9–5, 9–10, 9–21
menu, 9–14, 9–20, 9–21
momentary normally
  closed push button, 9–10, 9–21
  open push button, 9–10, 9–21
moving, 9–39
multi-component, 9–51
multistate indicator, 9–5, 9–9, 9–21
multistate push button, 9–5, 9–10, 9–21
non-retentive, 9–3, 9–4
normally closed momentary push button, 9–5
normally open momentary push button, 9–5
numeric data display, 9–5, 9–9, 9–21
numeric entry keypad, 9–5, 9–9, 9–13
numeric input cursor point, 9–5, 9–9, 9–22
numeric keypad-enable button, 9–6, 9–10, 9–22
overlapping, 9–42
panels, 9–6, 9–12, 9–22, 9–33
pasting, 9–36
planning addresses, 2–5
read, 9–2
rectangles, 9–6, 9–12
redo, 9–19
retentive, 9–3, 9–4
  addressing, 9–4
  safety considerations, 9–4
return to previous screen button, 9–6, 9–11, 9–21, 9–22
scale, 9–6
scales, 9–11
screen keypad enable button, 9–6, 9–11, 9–21
screen list selector, 9–6, 9–11, 9–21
screen print button, 9–6, 9–11, 9–21
screen select keypad, 9–6, 9–11
scrolling list, 9–6, 9–11
selecting, 9–15
set bit cursor point, 9–6, 9–7, 9–21
sizing, 9–38
spacing, 9–45
states
  configuring, 9–55, 9–56, 9–57
  viewing, 9–56
static, 9–2
symbols, 9–6, 9–12
text, 9–6, 9–12, 9–21, 9–24
time display, 9–6, 9–13, 9–21
toolbox, 9–20
touch screen, 9–1
trend, 9–6, 9–21
types, 9–1
undo, 9–19
ungrouping, 9–40
wedge, 9–6, 9–12, 9–21, 9–33
working with, 9–1
write, 9–2
zooming, 9–18

Offlink address, 2–39, 5–6, 5–19
Offset, 2–23, 2–47
Online help, P–3, P–10
Opening
  applications, 3–3
  screens, 8–3
    multiple, 8–3
Operation hold time, 12–7
Operations, in expressions, 7–4
  arithmetic, 7–9
  comparison, 7–11
  construct, 7–12
  logical, 7–16
Optional write expression, 7–2
OR, in expressions, 7–17
Organizing, tags, 6–4
Overlapping, objects, 9–42

P

PanelBuilder 1200 applications, converting, 3–4
PanelBuilder 1400e
  about, 1–1, 1–2
  components, 1–1
  default file format, 3–8
  dialog boxes, 1–1
  documentation, P–3
  downloading applications, 1–4
  editors, 1–1, 4–1
  equipment requirements, P–10
  features, P–1
  hardware requirements, P–10, 1–3
  optimizing performance, 2–4
  optional equipment, 1–6
  planning applications, 2–2
  README.TXT file, P–3
  related publications, P–6
  software registration, P–3
  software requirements, P–10, 1–3
  support services, P–10
  troubleshooting, D–1
  uploading applications, 1–4
  validation, errors, D–3
  version compatibility, 1–2
  wildcards, 6–3, 6–6
PanelBuilder 900, importing a database, 6–36
PanelBuilder for DOS applications, converting, 3–4
Panels, 9–6, 9–12, 9–22, 9–33
PanelView 1200 Enhancement Kit, 3–11
PanelView 1200/1400e Transfer Utility, about, 15–1
PanelView terminal
  enhancing, 1–2
  information message window, 14–2
  off-line to PLC, 11–47
  terminal address
    about, 5–18
    defining, 5–19
  types required, 1–2, 1–3
  version control at, E–4
Partial racks. *See* Racks, partial
Pasting, objects, 9–36
PB1400E.INI file, 3–13
PCMCIA memory card, 1–7
  file format, 3–8
Performance, optimizing, 2–4, 6–3
PID mnemonics, C–13
Planning
  addresses, 2–19
  applications, 2–1, 2–2
    ControlNet, 2–5
    DH+, 2–5
    Remote I/O, 2–5
    safety considerations, 2–2
  object addresses, 2–5
  rack assignments, 2–16
  screens, 2–5
  tags, 2–19
PLC controlled audio, 11–22
PLC controlled date and time control tag, 13–4
PLC controlled QTY/TIME reset, 11–46, 11–48
PLC controlled relay, 11–23
PLC controlled screen number tag, 13–3
PLC controlled screen print tag, 13–4
PLC controlled silence alarms, 11–22
PLC I/O control options
  about, 13–1
  date and time to PLC controller data tags, 13–6
  PLC controlled date and time control tag, 13–4
  PLC controlled screen number tag, 13–3
  PLC controlled screen print tag, 13–4
  screen number to PLC controller tag, 13–6
  screen print active to PLC controller tag, 13–6
PLC station address

about, 5–4, 5–5
defining, 5–7

PLC type, 5–22

PLC-2, addressing syntax, C–1

PLC/Scanner Type, recording on worksheets, 2–31

PLC-3, addressing syntax
data table sections, C–2
timers and counters, C–2

PLC-5, addressing syntax, 2–37, C–1
binary files: optional syntax, C–3
data files, C–3
I/O and status files, C–4
timers, counters and control files, C–4

PLC-5 (Enhanced), addressing syntax, 2–37, C–1
binary files: optional syntax, C–6
binary, integer, BCD, ASCII, float and string files, C–5
I/O and status files, C–5
timer, counter, control, SFC status, message, PID, block transfer and token data files, C–6

Powerup screen. *See* Application startup screen

Precedence, order of, in expressions, 7–4, 7–9, 7–11, 7–12, 7–16

Print conventions, P–7

Printer
compatible with development computer, 1–7
compatible with PanelView, 1–7
setting up, 3–15
solving problems, 3–15

Printing, screen reports, options, 3–14

Private tag databases
about, 6–30
converting to shared, 6–33

Problems, D–1
validation, D–3

Programmable controllers
and PanelView terminals, 2–13
for DH+ applications, 2–37
I/O image table, 2–28
PLC addressing syntax, C–1
reducing addressing requirements, 2–18
safety considerations, 2–3
SLC addressing syntax, C–7
software requirements, 1–6
types supported, 1–5
PLC types, 1–5
scanners, 1–5
user manuals, P–6

Publications, related, P–6

Push button hold time, setting, 12–6

Push buttons, 9–10
changing touch margins, 9–50

Pyramid Integrator, bridging DH+ networks, 2–40

Q

QTY/Time reset button, 11–48
enabling, 11–49

QTY/TIME reset to PLC, 11–48

R

Rack Assignments Worksheet, A–1
using, 2–31

Racks
access
full, 2–16, 5–9
listen only, 2–16, 5–9
assigning, 5–8, 5–9
assigning on worksheets, 2–32
changing, 5–14
defining, 5–9
deleting, 5–14
full, 2–13
Last chassis, 2–16, 5–10
minimizing scan time, 2–15, 2–16
minimum size, 2–18
module groups, 2–14
partial, 2–13, 2–32
planning, 2–16
shared, 2–15
size requirements, 2–18

Read objects, 9–2

README.TXT file, P–3

Rectangles, 9–6, 9–12

Redo, 9–19

Ref No., 2–45

Regeneration, of expressions, 7–20

Registering PanelBuilder 1400e, P–3

Related publications
I/O scanner modules, P–6
programmable controllers, P–6

Relaxed validation, 6–21

Remote alarm parameters, setting, 12–7

Remote I/O
and PanelView, 1–5
choosing as data source, 6–3, 6–23
reducing scan time, 2–15

setting up terminal communications, 5–3
Remote I/O Scanner, user manuals, P–6
Remote I/O Tags Worksheet, A–1
recording tag information, 2–22
using, 2–21
Remote keyswitch, 1–7
Remote station address, 2–38
Removing, graphic images, 9–28, 9–31
Removing filters, in Tag Usage Viewer, 6–28
Renaming
applications, 3–10
screens, 8–5
tags, 6–30
Repairing, shared database, 6–35
Reports
creating, 3–12
options, 3–13
saving as ASCII file, 3–16
screen, 3–14
setting up printer, 3–15
solving printing problems, 3–15
Required equipment, 1–3
Reserved keywords, F–1
Reset QTY/TIME button, 11–46
Resetting, the alarm status screen, 11–46
Retain cursor on cancel, configuring, 12–10
Retentive objects, 9–3
addressing, 9–4
safety considerations, 9–4
Return to previous screen button, 9–6, 9–11, 9–21
creating, 9–22
Return values, in expressions, 7–19
RIO. *See* Remote I/O
RIO Block Transfer Files editor, 5–2, 5–11
RIO Racks editor, 5–2, 5–8
RM, modules, 2–40
RS-232 port, 1–7
RSView, and sharing tag databases, 6–32, 6–34
Running, applications, 1–2

S

Safety considerations, 2–2
for retentive objects, 9–4
programming the controller, 2–3
Sample expressions, 7–28
Saving
applications, 3–6, 3–7
in another file format, 3–8
messages. *See* Updating messages
screens. *See* Updating screens
Scale, 2–23, 2–47, 9–6
Scales, 9–11
Scan classes, configuring, 5–24–5–25
for Device data source tags, 5–24
Scan Classes editor, using, 5–24–5–25
Scanning
about, 2–42
scan classes, 2–42
configuring, 2–42, 5–24
scan period, 2–42
background, 2–42, 5–25
foreground, 2–42, 5–25
Screen keypad enable button, 9–6, 9–11, 9–21
Screen list selector, 9–6, 9–11, 9–21
Screen number to PLC controller tag, 13–6
Screen print active to PLC controller tag, 13–6
Screen print button, 9–6, 9–11, 9–21
Screen prints
from the terminal, 13–4
PLC controlled screen print tag, 13–4
screen print active to PLC controller tag, 13–6
Screen reports, options, 3–14
Screen select keypad, 9–6, 9–11
Screen selectors, 9–11
Screen templates, designing, 2–7
Screen worksheets
for 1000e Terminals, 2–10, A–1
for 1200e Terminals, 2–8, A–1
for 1400e Terminals, 2–10, A–1
Screens
assigning application startup screen, 8–15
assigning new screen number, 8–2
assigning security, 8–8
closing, 8–7
color palette, 9–13
copying, 8–3
controls, 8–5
screen contents, 8–4
creating, 8–1
deleting, 8–7
designing, 2–6
grids, 8–11
opening, 8–3

planning, 2–5
PLC controlled screen number tag, 13–3
renaming, 8–5
saving. *See* updating
screen number to PLC controller tag, 13–6
setting background color, 8–7
templates, 8–3
toolboxes, 9–13
updating, 8–6
using templates to design, 2–7
using worksheets to design, 2–8
validating, 8–9
viewing tags used, 8–10

Scrolling list, 9–6, 9–11
cursor list, 9–59
object lists, 9–59
defining number of states, 9–59

Searching for text, in the tag usage viewer, 6–28

Security, assigning to screens, 8–8

Select
command, 9–21
tool, 9–21

SELECT CASE, in expressions, 7–15

Selecting
object components, 9–17
objects, 9–14, 9–15
shared database, 6–34
tags, 6–6
terminal type, 12–2

Selection conventions, P–7
dialog boxes, P–8
shortcut keys, P–9

Serial upload/download cable, 1–6

Set bit cursor point, 9–6, 9–7, 9–21
retain cursor on cancel, 12–10

Setting
auto-repeat functions, 12–8
baud rate, remote I/O, 5–3
ControlNet station number, 5–17
DH+ station number, 5–17
push button hold time, 12–6
remote alarm parameters, 12–7
target firmware version, 12–4

Setting up
block transfer files, 5–10
network file transfers
ControlNet, 5–18
DH+, 5–18
Remote I/O, 5–4
terminal communications
ControlNet, 5–16
DH+, 5–16
Remote I/O, 5–3

SFC status mnemonics, C–15

Shared racks, 2–15

Shared tag databases
about, 6–31
repairing, 6–35
selecting, 6–34
using, 6–32
with RSView, 6–32, 6–34

Shortcut keys, P–9

Signed integer data type, 2–24, 2–48

Silence alarms to PLC, 11–22

Single-line alarm message window, 11–30
choosing, 11–36

Sizing
list objects, 9–59
objects, 9–38

SLC 5, addressing syntax, 2–37

SLC 5 (Enhanced), addressing syntax, 2–37, C–7
binary files: optional syntax, C–9
binary, integer, float, ASCII and string files, C–8
I/O files, C–7
status files, C–8
timer, counter and control files, C–9
analog member addressing: optional syntax, C–11
bit member addressing: optional syntax, C–10

Software registration, P–3

Sorting, entries in a spreadsheet, 4–10

Spacing, objects, 9–45

Specifying, device type, 5–9

Spreadsheets
changing column width, 4–13
changing row height, 4–13
copying selections, 4–9
cutting selections, 4–9
deleting data, 4–12
duplicating rows, 4–11
editing, 4–12
editing selections, 4–12
filling with identical data, 4–11
inserting rows, 4–12
pasting selections, 4–9
selecting a block of cells, 4–8
selecting a cell, 4–7
selecting a row, 4–7
selecting a single column, 4–8
selecting more than one column, 4–8
selecting more than one row, 4–8
selecting the entire spreadsheet, 4–8

sorting entries, 4–10
using, 4–3
using Undo and Redo, 4–7
working in, 4–6

Stacking. *See* Overlapping

Start address, 2–27

Startup screen. *See* Application startup screen

Static objects, 9–2
converting to wallpaper, 9–40

Station address, 2–37, 5–22
local, 2–38
remote, 2–38

Station number
ControlNet, 5–17
converting from ControlNet to DH+, 5–34
converting from DH+ to ControlNet, 5–32
DH+, 5–17

String tags, 6–2
configuring, 6–17

Support services, P–10

Symbols, 9–6, 9–12
creating, 9–26
exporting, 9–28

Syntax, in expressions
checking, 7–23
errors, 7–23

T

Tag data, specifying source, 6–21, 6–23

Tag database
and RSView, 6–1
defined, 6–1
editing tags, 6–11
importing and exporting, 6–35
organizing, 6–3
private, 6–30
converting to shared, 6–33
selecting tags, 6–6
shared, 6–30
repairing, 6–35
selecting, 6–34
Tag Database editor, 6–5

Tag Database editor
about, 6–4
editing tags, 6–11
using, 6–5
using the spreadsheet, 6–10
wildcards, 6–6

Tag folder, creating, 6–4

Tag spreadsheet viewer, 6–10

Tag start value, 2–50

Tag Usage Viewer, 3–11
filtering data, 6–26
for screens, 8–10
removing filters, 6–28
searching for text, 6–28
using, 6–25

Tags
adding to folders, 6–9
analog, 6–2
configuring, 6–13
in expressions, 7–4
and addresses, 2–13, 2–16, 2–19, 2–43
and Remote I/O, 6–2
assigning a name, 2–22, 2–46
assigning to objects, 9–61
block, 6–2
configuring, 6–18
configuring type, 6–13
converting, 5–26
converting from Remote I/O to ControlNet, 5–28–5–35
copying between applications, 8–4
creating, 6–1, 6–23
other methods, 6–23
data sources, 6–2, 6–20
data types, in expressions, 7–4
defined, 6–1
described, 2–19
digital, 6–2
configuring, 6–16
in expressions, 7–4
duplicating, 6–11
editing, 6–11
grouping with folders, 6–4
initial value, 2–27, 9–3
naming conventions, 6–4
PanelBuilder 1400e, 6–3, 6–6
nesting folders, 6–4, 6–8
opening multiple folders, 6–8
organizing, 6–3, 6–4
planning, 2–19
read tags, 6–2
recording the address, 2–27, 2–50
renaming, 6–30
scanning, 2–42
selecting in the tag database, 6–6
string, 6–2
configuring, 6–17
types, 2–23, 2–47, 6–2
updating values of, 5–24
using the Tag Usage Viewer, 6–25
viewing in a folder, 6–8
write tags, 6–2

Target firmware version, 12–4

defined, E–1
specifying, 12–4

Technical support services, P–10

Templates, 8–3
screen, 2–7

Terminal setup options
about, 12–1
configuring, 12–1

Terminals
selecting type, 12–2
types required, 1–2, 1–3

Testing, applications, 2–2

Text, 9–6, 9–12, 9–21
changing appearance, 9–50
creating, 9–24
editing, 9–25
extended characters, 9–25
inner text, 9–24
arranging, 9–46
labels, 9–34
sizes, 9–50
underlining, 9–50

Time and date, in PanelView terminal
date and time to PLC controller data tags, 13–6
PLC controlled date and time control tag, 13–4

Time display, 9–6, 9–13, 9–21

Timeout, 2–38, 5–23

Timer mnemonics, C–12

Timing parameters, 12–5
push button hold time, 12–6
remote alarm, 12–7

Token data mnemonics, C–15

Toolboxes
items, 9–14, 9–21
using, 9–13, 9–20

Touch margins, 9–50

Touch screen objects, 9–1

Transfer inhibit control, 5–7, 5–20

Transfer request control, 5–7, 5–20

Transfer status control, 5–7, 5–20

Transferring applications
about, 15–1
from within PanelBuilder, 15–1
using the PanelView 1200/1400e Transfer Utility, 15–3

Trend, 9–6, 9–21

Trigger value
and messages, 10–2
simplifying PLC programming, 10–2

Triggering alarms, 11–2

Troubleshooting, D–1
validation, D–3

Type, 2–45

U

Undo, 9–19

Ungrouping
nested groups, 9–41
objects, 9–41, 9–42

Unsigned integer data type, 2–24, 2–48

Unsolicited messages, about, 5–23

Updating
messages, 10–14
screens, 8–6

Upload/download cable, 1–6

Uploading applications, 15–1

Usability, and screen design, 2–6

User entry character (”?”), in expressions, 7–6
in write expressions, 7–24

V

Validating
applications, 3–10
relaxed validation, 6–21
expressions, 7–27
messages, 10–13
problems with, D–3
screens, 8–9
tag definitions, 6–21

Validation results window, 3–11, 8–9, 10–13

Variables, local
evaluation of, 7–7
in expressions, 7–6
naming, 7–6

Version 1.03 firmware features, E–2

Version 2 firmware features, E–2

Version 3 firmware features, E–3

Version 4 firmware features, E–3

Version compatibility, 1–2

Version control
and downloading applications, E–3
at the PanelView terminal, E–4
defined, E–1
target firmware version, 12–4

Viewing
forms, 4–4

spreadsheet, 4–4
tags in a folder, 6–8
tags used in a screen, 8–10

W

Wallpaper, 9–40

Wedge, 9–6, 9–12, 9–21, 9–33

Wildcards, 6–3, 6–6

Word-triggered alarms, 11–6
about, 11–8

Worksheets
about, A–1
completing, for a Remote I/O application, 2–20, 2–35
recording addresses, 2–27, 2–50
sample Worksheets, A–1
1000e Terminals, 2–10, A–1
1200e Terminals, 2–8, A–1
1400e Terminals, 2–10, A–1
Block Transfer File, 2–33, A–1
Block Transfer File I/O Usage, 2–29, A–1
ControlNet Communications, 2–36, A–1
Device Tags, 2–43, A–1
DH+ Communications, 2–36, A–1
Discrete I/O Usage, 2–28, A–1
Rack Assignments, 2–31, A–1
Remote I/O Tags, 2–21, A–1
types, 2–5
using to design screens, 2–8, 2–10
using to help plan applications, 2–5
using to plan addresses, 2–19

Write expressions, 7–24
assigning, 7–25

Write objects, 9–2

X

XOR, in expressions, 7–17

Z

Zooming, 9–18

Rockwell Automation

Rockwell Automation helps its customers receive a superior return on their investment by bringing together leading brands in industrial automation, creating a broad spectrum of easy-to-integrate products. These are supported by local technical resources available worldwide, a global network of system solutions providers, and the advanced technology resources of Rockwell.

Worldwide representation.

Argentina • Australia • Austria • Bahrain • Belgium • Bolivia • Brazil • Bulgaria • Canada • Chile • China, People's Republic of • Colombia • Costa Rica • Croatia • Cyprus Czech Republic • Denmark • Dominican Republic • Ecuador • Egypt • El Salvador • Finland • France • Germany • Ghana • Greece • Guatemala • Honduras • Hong Kong Hungary • Iceland • India • Indonesia • Iran • Ireland • Israel • Italy • Jamaica • Japan • Jordan • Korea • Kuwait • Lebanon • Macau • Malaysia • Malta • Mexico • Morocco The Netherlands • New Zealand • Nigeria • Norway • Oman • Pakistan • Panama • Peru • Philippines • Poland • Portugal • Puerto Rico • Qatar • Romania • Russia • Saudi Arabia • Singapore • Slovakia • Slovenia • South Africa, Republic of • Spain • Sweden • Switzerland • Taiwan • Thailand • Trinidad • Tunisia • Turkey • United Arab Emirates United Kingdom • United States • Uruguay • Venezuela

Rockwell Automation Headquarters, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414 382-2000 Fax: (1) 414 382-4444
Rockwell Automation European Headquarters, Avenue Hermann Debroux, 46, 1160 Brussels, Belgium, Tel: (32) 2 663 06 00, Fax: (32) 2 663 06 40
Rockwell Automation Asia Pacific Headquarters, 27/F Citicorp Centre, 18 Whitfield Road, Causeway Bay, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846
World Wide Web: http://www.ab.com

Publication 2711E–819 – January 1998
Supersedes Publication 2711E–819 – July 1997
40061-296-01(D)
Copyright 1998 Allen-Bradley Company, Inc. Printed in USA